Deep Learning-Based Lung Cancer Detection and Classification Using CT Scan Image

Alemu Gudeta Eticha

A Thesis Submitted to the Department of Computer Science and Engineering

School of Electrical Engineering and Computing

Office of Graduate Studies

Adama Science and Technology University

October, 2024

Adama, Ethiopia

Deep Learning-Based Lung Cancer Detection and Classification Using CT Scan Image

Alemu Gudeta Eticha

Advisor: Teklu Urgessa (PhD)

A Thesis Submitted to the Department of Computer Science and Engineering

School of Electrical Engineering and Computing

Office of Graduate Studies

Adama Science and Technology University

October, 2024

Adama, Ethiopia

Declaration

I hereby declare that this Master thesis entitled "**Deep Learning-Based Lung Cancer Detection and Classification Using CT Scan Image**" is my original work. That is, it has not been submitted for the award of any academic degree, diploma, or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged through citation.

_____	_____	_____
Name of student	Signature	Date

Recommendation

I, the advisor of this thesis, hereby certify that I have read the revised version of the thesis entitled "**Deep Learning-Based Lung Cancer Detection and Classification Using CT Scan Image**" prepared under my guidance by **Alemu Gudeta Eticha** submitted in partial fulfillment of the requirements for the degree of Master of Science in Computer Science and Engineering. Therefore, I recommend the submission of a revised version of the thesis to the department following the applicable procedures.

_____	_____	_____
Advisor	Signature	Date

Approval sheet

I, the advisor of the thesis entitled "**Deep Learning-Based Lung Cancer Detection and Classification Using CT Scan Image**" and developed by Alemu Gudeta Eticha, hereby certify that the recommendations and suggestions made by the board of examiners are appropriately incorporated into the final version of the thesis.

_____	_____	_____
Advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the thesis by Alemu Gudeta Eticha have read and evaluated the thesis entitled "**Deep Learning-Based Lung Cancer Detection and Classification Using CT Scan Image**" and examined the candidate during open defense. This is, therefore, to certify that the thesis is accepted for partial fulfillment of the requirement of the degree of Master of Science in Computer Science and Engineering.

_____	_____	_____
Chairperson	Signature	Date
_____	_____	_____
Internal Examiner	Signature	Date
_____	_____	_____
External Examiner	Signature	Date

Finally, approval and acceptance of the thesis is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

_____	_____	_____
Department Head	Signature	Date
_____	_____	_____
School Dean	Signature	Date
_____	_____	_____
Office of Postgraduate Studies	Dean Signature	Date

Acknowledgements

I am grateful to **God** for His unwavering support, which has been my guiding light through the highs and lows of this journey. His constant presence has given me the strength to persevere, especially during moments of uncertainty. Moreover, I want to extend my heartfelt thanks to my advisor, **Dr. Teklu Urgessa**, for his invaluable support and encouragement throughout my research. His thoughtful guidance and advice have been instrumental in helping me navigate the complexities of my work.

I am also thankful to **Mr. Abera Fikre**, the radiologist at Adama Hospital, for his assistance with data annotation and labelling. His expertise was crucial in ensuring the reliability and validity of my findings.

Lastly, I want to express my deep appreciation to my family and friends for their unwavering support and encouragement. Their belief in me has provided the strength to face challenges head-on. I truly appreciate everything they have done for me. This journey would not have been possible without the incredible support of all these wonderful people.

Table of Contents

List of Figures

List of Tables

List of Acronyms

ACS	-	American Cancer Society
Adamax	-	Adaptive moment estimation with Maximum
AI	-	Artificial Intelligence
API	-	Application Programming Interface
CAD	-	Computer Aided Diagnosis
CAM	-	Class Activation Map
CIA	-	Cancer Imaging Archive
CNN	-	Convolutional Neural Networks
CNN-LSTM	-	Convolutional Neural Networks and Long Short-Term Memory
CNTK	-	Microsoft Computational Network Toolkit
COVID	-	Coronavirus Disease
CT	-	Computed Tomography
DICOM	-	Digital Imaging and Communications in Medicine
DNA	-	DeoxyriboNucleic Acid
FGFR	-	Fibroblast Growth Factor Receptor
HIV	-	Human immunodeficiency virus
GAWA	-	Genetic Algorithm with Wrapper Approach
GLCM	-	Gray-Level Co-occurrence Matrix
GradCAM	-	Gradient-Weighted Class Activation Map
IQ-OTH/NCCD	-	Iraq-Oncology teaching hospital/national center for cancer diseases
LDCT	-	Low Dose Computed Tomography
LIDC/IDRI	-	Lung Image Database Consortium & Image Database Resource Initiative
LIME	-	Local Interpretable Model-Agnostic Explanations
LRP	-	Layer-wise Relevance Propagation
LUNA16	-	Lung Nodule Analysis 2016 dataset
MRI	-	Magnetic Resonance Imaging
NIHR	-	National Institute for Health and Care Research

NSCLC	-	Non-Small Cell Lung Cancer
PET	-	Positron Emission Tomography
ReLU	-	Rectified Linear Unit
ROI	-	Regions of Interest
SCLC	-	Small Cell Lung Cancer
SHAP	-	SHapley Additive exPlanations
SVM	-	Support Vector Machine
TASH	-	Tikur Anbessa Specialized Hospital
TNM	-	Tumor, Nodes and Metastases
VGG	-	Validation of Goodness of Genetic Variants
WHO	-	World Health Organization
XAI	-	Explainable artificial intelligence

Abstract

Early detection is significant for improving lung cancer patient outcomes. However, risk concerns and resource scarcity bring challenges, especially in developing nations. This study offers a method for precise and effective lung cancer detection utilizing deep learning techniques. The black-box nature of deep neural networks challenges its use in mission-critical applications, raising ethical and moral concerns inducing a lack of trust. Explainable Artificial Intelligence and Result-based visualization techniques are methods used to add clinical trust to the decision logic of a deep learning models. In this study, four Pre-trained deep learning models are utilized for feature extraction from computed tomography images, optimizing computing complexity. Lung CT data was gathered from Tikur Anbessa Specialized Hospital in Addis Ababa, Ethiopia, to validate the proposed approach. After data cleaning is applied, 3500 CT images are prepared where 2000 images are cancerous and 1500 images are normal. Data augmentation is employed to improve the training performance by increasing training data size. For both classes, 90% of the dataset is used for training, while 10% is used for validation. In this study, we used GradCAM and LIME to visualize the detection's region of interest and detection logic. GradCAM is a visualization tool that demonstrates special features that lead a convolutional neural network to generate its decisions on a target image, while LIME is a technique that explains the predictions of any classifier by approximating it locally with an interpretable model, thereby providing insights into the specific features influencing the model's decisions. Utilizing GradCAM and LIME, our study achieved a promising result in visualizing and comprehending the features that affect the model's detection, overcoming the black-box nature of deep learning. The outcomes of this study offer an opportunity to substantially improve the detection of lung cancer. As a result, the outperforming Densenet169 obtained 94.85% accuracy, 93.42% precision, 94.67% recall, and a 94.04% F1 score. As future work, we plan to improve data size for robust model training and better performance.

Key Words: *Lung Cancer, Computed Tomography, Deep Learning, DenseNet169, InceptionV3, Xception, ResNet50, LIME, GradCAM, Explainability*

Chapter one

1. Introduction

1.1 Background of the study

Lungs are air-filled organs within the thoracic cavity (chest) and constitute the main part of the human respiratory system. Lung cancer, which accounts for about 27% of cancer-related deaths, is the most common cause of cancer-related deaths (Schabath and Cote 2019). Permanent alterations in deoxyribonucleic acid (DNA) sequences are referred to as mutation, or uncontrolled tissue growth. Mutations may result from inherited defects in genetics or external factors. Tobacco smoke chemicals are the most common external factor, although there are many other carcinogens as well. However normal circumstances, injured tissue regenerates itself, when a malignant mutation develops, the new tissue grows uncontrollably, giving rise to cancerous cells. So, Lung nodules are abnormal growths within the lung that can be malignant or benign (Shafi et al. 2022).

Lung cancer starts when abnormal cells grow in an uncontrolled way in the lungs. It is a serious health issue that can cause severe harm and death. According to a survey by the WHO, Lung Cancer was the fourth leading of death in 2019 globally, and is responsible for the passing away of about 1.8 million people every year (WHO, 2019). Even though Only 20% of lung cancer cases are reported to occur in low and middle income countries, an estimated 1.5% of all Ethiopian cancers involve the lung (Gebremariam et al. 2021) and Lung cancer is the third most lethal cancer in Ethiopia, next to leukemia and prostate cancer according to (Awedew, Asefa, and Belay 2022).

Moreover, there are six stages of lung cancer, which range from stage 0 (carcinoma in situ) to stage IV (metastatic disease). The stage of lung cancer influences treatment choices and establishes the amount of the disease's dissemination, with earlier stages providing a higher likelihood of effective intervention (WHO, 2023). The cancer cells move from the lungs into the lymph nodes and ultimately the blood circulation. Due to the obvious natural movement of lymph, cancer cells usually spread towards the middle of the chest. In this scenario, early detection and classification can help preventing metastasis, which would happen if this cancer had spread to other organs (Rajasekar et al. 2023).

For lung cancer is mostly exposed at the later stages, early-stage lung cancer can be challenging to detect, especially considering the expensive cost of computed tomography (CT) and the frequent radiation exposure it causes. Even for experts, reviewing CT scans of the lungs to find pulmonary nodules - particularly those indicative of cell lung cancer is a laborious and error prone process (Shafi et al. 2022).

The ability of Computer Aided Diagnosis (CAD) to detect lung cancer in its early stages has been greatly improved by the development of modern imaging technologies and deep learning algorithms. These methods use complex algorithms to examine X-rays, CT scans, and MRIs, among other medical images. This allows for the detection of minute anomalies that could go unnoticed by humans. CAD has the potential to enhance patient outcomes and prognosis by enabling early detection, fast intervention, and more effective treatment techniques (R. Tekade et al., 2018). By incorporating CAD into clinical practice, medical professionals may be able to diagnose lung cancer more consistently and objectively and with fewer diagnostic errors and variability. It is anticipated that these systems will become more and more important in the early diagnosis and treatment of lung cancer as CAD research and development advance, ultimately helping to lower the death rates linked to this miserable illness(Venkatesh et al. 2023).

Deep learning is an extremely promising strategy for the problems associated with lung cancer detection (Humayun et al. 2022). It can uncover complex patterns and attributes from enormous amounts of medical image data that may be suggestive of lung cancer. These algorithms can acquire the high performance necessary to detect minor abnormalities and discriminate between Cancerous and Normal lung by training on extensive datasets of CT scan images (Venkatesh et al. 2023). Furthermore, our study considers the idea of explainable Artificial Intelligence and result-based detection's area visualization to overcome the doubt of clinical practitioners for Artificial intelligence-based detection. Grad-CAM and LIME based Visualization is used to visualize and enhance the model's performance on identifying features led to detection.

So, in our study, we revealed the ability of deep learning algorithms to analyze CT scan images for lung cancer detection for further treatment with improved clinical trust. Moreover, we experimented with four Pre-trained Deep Learning models which are trained on huge amount of data, to identify the better architecture to detect lung cancer with a better model evaluation result and minimized computational complexity.

1.2 Motivation of the study

The urgent need to address Ethiopia's rising lung cancer rate (Gebremariam et al. 2021) - a developing nation with inadequate healthcare resources and high lung cancer death rates is the driving force behind this study. Ethiopia is not an exception to the global trend of lung cancer becoming a major public health concern due to increased incidence rates and a scarcity of extensive early screening operations. Reducing mortality and improving patient outcomes are dependent on the early detection of lung cancer. As a useful diagnostic technique, computed tomography (CT) scan imaging has proven to be effective in scanning nodules of lung cancer in its early stages (Sfeir, Agnel and Symington 2017)**.**

However, healthcare practitioners may find it difficult and time-consuming to interpret CT scan images (Schabath and Cote 2019), especially in countries with limited assets like Ethiopia. Thus, it seems practical to create an automated and precise lung cancer detection model that would help medical professionals detect the disease at early stages by utilizing deep learning-based techniques.

1.3 Statement of the Problem

While existing deep learning models excel in lung cancer detection, their lack of detection transparency and Explainability hinders clinical trust and limits personalized treatment options. This research bridges the transparency gap by integrating domain-specific model's decision transparency and Explainability into a deep learning model. Our approach unveils the model's reasoning, revealing key features and relationships driving its detection. This enhanced transparency fosters clinician trust, facilitates accurate diagnosis through focused analysis, and paves the way for personalized treatment plans tailored to the specific features identified by the model. (Venkatesh et al. 2023) could achieve an accuracy of 99.6% in detection of lung cancer, but the clinical trust of their model is not addressed.

By integrating Explainable techniques, i.e., LIME and GradCAM, our approach unlocks a deeper understanding of our model's region of interest and reasoning for lung cancer detection. This transparency can build trust among clinicians, enhance diagnostic accuracy through focused analysis of relevant features. Also, the gap related with computational complexity with previous studies, which is improved introducing the concept of transfer learning, and model performance improvements are concerns behind our study.

1.4 Research questions

The study that will be conducted attempts to answer the following questions:

RQ-1: Which Pre-trained deep learning models are suited for lung cancer detection?

RQ-2: What techniques are used to reduce computational complexity of deep learning models in lung cancer detection?

RQ-3: To what extent we can improve both disease detection's performance and transparency?

1.5 Objectives

1.5.1 General Objectives

The general objective of this study is to develop transparent, light-weight lung cancer detection model using computed tomography image.

1.5.2 Specific Objectives

The research follows the specific objectives listed below to achieve the general objective.

- To analyze an ideas and related studies about lung cancer detection.
- To collect lung CT image data, prepare and augment the dataset to enhance its diversity and robustness by introducing the data augmentation technique
- To identify the suitable pre-trained CNN model for developing lung cancer detection.
- To train the model with the prepared datasets and assess its performance using a test dataset and evaluation metrics
- To implement result-based visualization techniques to improve lung cancer's detection understandability.

1.6 Significance of the study

Due to its straightforward advantages to patients and healthcare professionals, this study is extremely important in the field of healthcare. By using deep learning algorithms and computed tomography (CT) scan images, this study establishes lung cancer detection that will give support in quick and reliable diagnoses. For patients, it's particularly helpful since early detection of lung cancer is critical to better treatment outcomes and overall survival rates. By enabling quick intervention and treatment means, our proposed detection model could assist in the early diagnosis of lung cancer. This could result in more effective medications and better patient outcomes.

On the other hand, the research provides a reliable and efficient method for lung cancer diagnosis, which is of great benefit to healthcare providers. By giving medical professionals better decision-making skills, this research helps to improve patient care by enabling them to offer patients timely and effective interventions. The detection method and disease recognition transparency feature provide objective and standardized outcomes by analyzing and interpreting complex CT scan images using deep learning techniques. Furthermore, because the technology is more applied nowadays, it can speed up the diagnosis procedure, giving medical professionals more time to concentrate on other important aspects of patient care by detecting whether their lung is cancerous or normal.

1.7 Scope and limitations

1.7.1 Scope of the Study

The proposed model is capable of accurately detecting the occurrence and absence of lung cancer. The aim is to provide medical professionals with a powerful tool that enables the offer of high-quality healthcare services to patients or individuals seeking information about their lung cancer status. By leveraging deep learning algorithms and analyzing computed tomography (CT) scan images, the study seeks to create a model that can offer valuable insights into lung cancer occurrence, thereby facilitating early detection in small computation time and with better model performance and offering transparency using Explainable Artificial Intelligence technique – GradCAM and LIME.

Chapter Two

2. Literature Review

2.1 Overview of Lung Cancer

These days, one of the leading causes of death for young people worldwide is cancer. Among the most common malignancies identified in both men and women are stomach, prostate, lung, and breast cancers. If not detected early on, these diseases can cause serious complications or even death (Humayun et al. 2022). Cancer in the human body represents the abnormal growth of cells. Studies have shown that a healthy individual can be impacted by nineteen distinct types of cancer. The malignancy with the highest death rate among these is lung cancer (Schabath and Cote 2019). It is estimated that about 1.8 million people die annually due to this disease (WHO, 2023). Smoking has been identified as the primary cause of lung cancer, accounting for around 80% of all instances of the disease worldwide. Without specialized computer-aided diagnosis, lung cancer is difficult to identify in its early stages. (Singh and Gupta 2019).

About 25% of individuals with early-stage lung cancer had no symptoms at all, according to the studies. Lung cancer is not detectable to the naked eye, unlike other types of cancer, and its symptoms might be misinterpreted for those of other illnesses such coughing, bronchitis, and asthma. When a patient has a chest CT scan or X-ray for any other valid cause, lung cancer is often detected (Dimililer, Ugur, and K. Ever 2017). When the remaining 75% of individuals exhibit or acquire symptoms, a diagnosis is made, according to this study. These symptoms may be carried on by the primary tumor's direct effects, cancer's metastases, which have spread to other parts of the body, or problems with the hormones, blood, or other systems.

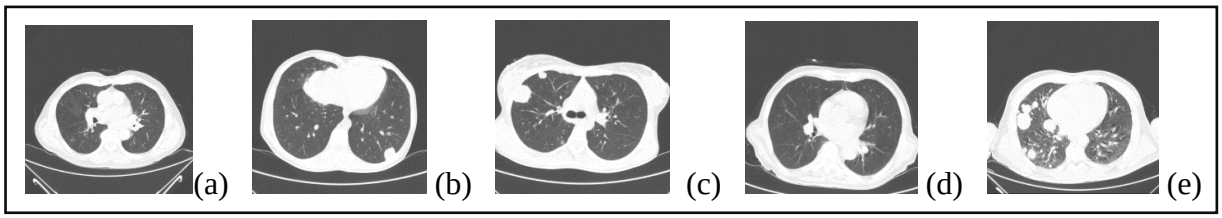


Figure 2.1 Lung Cancer screening sample

In Figure 2.1 above, (a) and (d) are normal lung whereas (b), (c) and (e) are malignant lung (cancerous), according to radiologist annotators.

2.1.1 Types of Lung Cancer

According to (Travis 2011), there are two main types of lung cancer. These are small cell lung cancer (SCLC) and non-small cell lung cancer (NSCLC). They are categorized as Figure 2.2.

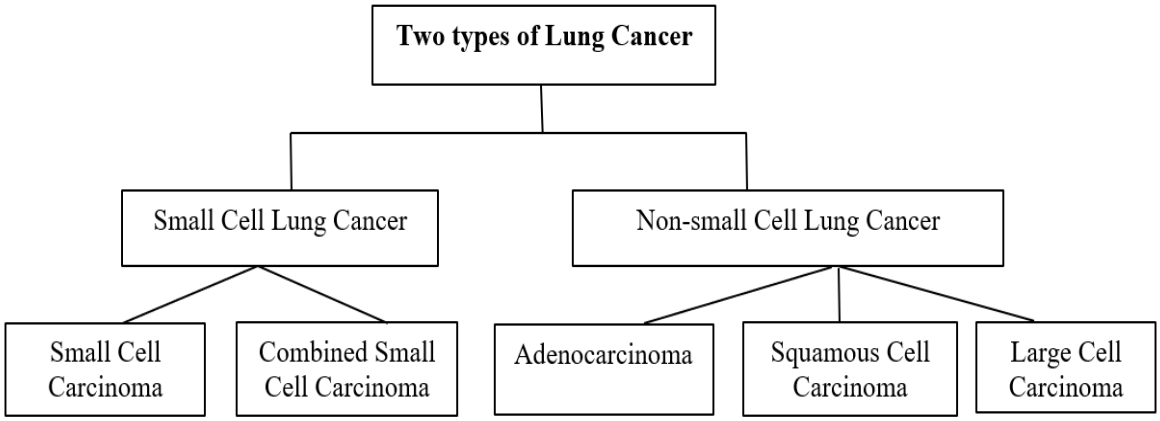


Figure 2.2 Types of Lung Cancer (Travis 2011)

1. Small Cell Lung Cancer (SCLC)

The various kinds of small cell lung cancer are called after the cell types that are present in the malignancy and the characteristics of these cells under a microscope. A rare kind of lung cancer that grows quickly is small cell lung cancer. Cigarette smoking is nearly always linked to small cell lung cancer. Although anyone can get it, those who have smoked tobacco for a long time usually do. According to American Cancer Society, giving up cigarettes is the only way to avoid small cell lung cancer (ACS 2024). SCLC is further classified as small cell carcinoma and Combined small cell Carcinoma (Mixed of small cell cancer and large cell cancer).

Small-cell carcinomas account for 10-15% of lung cancer cases. It can be classified as having limited disease when it only affects one lung and possibly one lymph node close to the lungs, or as having extensive disease when it affects both lungs and has progressed to the pleura or other organs (ACS 2024). WHO defines (C-SCLC as small-cell lung cancer (SCLC) plus additional components made up of any histological type of non-small-cell lung cancer (NSCLC); these are typically adenocarcinoma (ADC), squamous-cell carcinoma (SCC), and large-cell carcinoma (WHO, 2023).

2. Non-Small Cell Lung Cancer (NSCLC)

Like other malignancies, non-small cell lung cancer (NSCLC) starts at the cellular level and results in the fast and uncontrollable reproduction of abnormal lung cells. They are carcinomas of the cells lining the surface of the lung airways like bronchi, bronchioles, and alveoli are among them (ACS 2024). Smoking is the main risk factor for non-small cell lung cancer (NSCLC). The higher smoking habit and the earlier started, the higher chance of developing non-small cell lung cancer (NSCLC). Additional risk factors include HIV infection in the past, passive smoke, radiation exposure, environmental pollution, exposure to toxins at work, and family history of lung cancer.

Based on the kind of cells that are cancerous, there are three main kinds of non-small cell lung cancer (Minna, Roth, and Gazdar 2002). The first one is Adenocarcinoma, which accounts for 40% of the diagnosis for every NSCLC case. Both non-smokers and smokers can develop adenocarcinoma. Usually, adenocarcinoma starts in the mucus-producing cells that border the bronchioles, the tiny airways of the lung. Compared to other forms of lung cancer, adenocarcinoma often grows more slowly, which can improve prognosis. The second type of NSCLC is squamous cell carcinoma, covering 25% to 35% of all NSCLC diagnoses, also known as epidermoid carcinoma, is the second most prevalent kind of NSCLC. There are squamous cells in the lining of the bronchi, which are thin, flat cells that line the surfaces of organs. Treatment for these malignancies is more challenging since they are more likely to spread to other parts of the body. Smoking is the primary cause of squamous cell carcinoma compared to other forms of lung cancer (ACS 2024). The third type of NSCLC is Large cell carcinoma - a rare form of NSCLC, accounting for only 10% to 15% of all diagnoses. It can occur anywhere in the lung and tends to be aggressive. (WHO, 2023)

We made sure to include a wide range of lung cancer kinds in our study by thoroughly examining each one. We sought to capture the complexities and subtleties of lung cancer biology and treatment responses by incorporating a variety of histological subtypes, such as small cell lung cancer (SCLC) and non-small cell lung cancer (NSCLC), offering a more comprehensive knowledge of the disease's impact on patients.

2.1.2 Stages of Lung Cancer

The stage of lung cancer must be identified in order for the doctor to decide on the best course of treatment for the patient. With its help, they might be able to assess the treatment's chance of success. As both SCLC and NSCLC has their staging concept, Small cell lung cancer has two stages, which are limited stage and extended stage. It is referred to as limited stage small cell lung cancer if the illness is contained to a single location on one side of the chest. Furthermore, radiotherapy may be employed as a single site of treatment for the cancer. This frequently suggests that the cancer is contained to a single lung. A large spread of the cancer across the lung, to the other lung, to lymph nodes on the opposite side of the chest, or to other parts of the body is referred to as extended stage lung cancer. A lot of medical professionals also view SCLC that has progressed to the fluid around the lung as extended or advanced stage. (Van der Straeten and Tasson 1985)

As (WHO, 2023), compared to small cell lung cancer, staging non-small cell lung cancer is more difficult. The clinical or pathologic stage of NSCLC can be used to describe the disease. Physicians may utilize CT scans to obtain images of the patient's internal organs in order to determine the cancer's clinical stage. To verify the diagnosis, a biopsy may be performed, wherein a tiny sample of the tumor's tissue is removed and examined under a microscope (Travis 2011). If cancer surgery is done, healthcare practitioners can look at the patient’s tumor and see cancer’s pathologic stage. This tells the healthcare practitioners how far the cancer has grown. The most popular technique for sizing NSCLC tumors is the TNM (Tumor, Nodes and Metastases) system, which has the digits x, 0, 1, 2, 3, or 4 after each letter. These include the tumor's centimeter width, location within the lung, presence of multiple tumors in the same lung, degree of airway obstruction preventing the patient from having pneumonia or lung collapse, and whether the tumor has spread to other organs or lymph nodes.

The first stage is occult (x), during which the patient coughs up mucus that may contain cancer cells. Neither a biopsy nor an imaging scan can reveal the patient's tumor. Another name for it is hidden cancer. Stage 0 NSCLC refers to a patient's very small tumor at the time of diagnosis, after the occult stage. Cancer cells have not yet metastasized outside of the lungs or into the deeper lung tissues. (Van der Straeten and Tasson 1985)

- **Stage I (Early Stage) Lung Cancer**

Stage I refers to lung cancer's second-earliest stage. It suggests that abnormal cells in the respiratory system are the source of cancer. Nevertheless, the tumor remains limited to the lungs and has not spread to lymph nodes. The majority of people have a five-year or longer lifespan, and most of them are treatable. The primary tumor is currently limited to the lungs and measures only three centimeters in diameter; it has not yet spread to any surrounding structures or the visceral pleura. (Van der Straeten and Tasson 1985).

- **Stage II Lung Cancer**

A diagnosis of stage II lung cancer occurs when a physician finds one or more tumors in a single lung. It may or may not have affected the nearest lymph nodes due to the cancer. However, it hasn't reached distant locations like the patient's bones or other organs. Based on the size of the tumor and the location of any affected lymph nodes, cancer can be categorized as either localized or regional. In order to eliminate stage II cancer, doctors often employ surgery. Stage II lung cancer might vary in the size and extent of the original tumor. It may have a diameter larger than three centimeters or it could invade on nearby structures such as the diaphragm, mediastinal pleura, main bronchus, or chest wall. But it hasn't spread to the lymph nodes in the mediastinal area or anywhere else. (Van der Straeten and Tasson 1985).

- **Stage III Lung Cancer**

Stage III lung cancer is in just one lung. It also only affects the organs, lymph nodes, and tissue surrounding the organs. The cancer has not spread or metastasized past that stage. For this reason, this stage is also referred to as locoregional sickness or locally advanced illness. In Stage III, the primary tumor's dimensions are subject to alter. It may have invaded the diaphragm, main bronchus, mediastinal pleura, and chest wall, among other nearby structures. (Travis 2011)

- **Stage IV Lung Cancer**

Stage IV refers to the last and most serious stage of cancer. In stage IV cancer, the tumor has spread from the lungs to distant organs such the liver, brain, or bones. Although there isn't always a cure for cancer, there are several treatments that can help manage the symptoms and slow the disease's advancement.

Not only did our study look at all types of lung cancer, but it also included several stages of the disease without considering them as separate groups. We observed that all stages were regarded as positive for the 'read' status of stored patient data within the framework of our investigation. With this method, we were able to evaluate the overall effects of lung cancer regardless of how far along it was, which improved our understanding of patient outcomes and the effectiveness of treatment.

2.2 Computed Tomography

Based on the analysis, lung cancer is the most commonly considered disease in medical field to diagnose it in its earlier stage (Jin 2021). Normally, lung cancer is manually predicted with the help of the disease symptoms (Brock et al. 2008) such as blood coughing, chest pain, shortage of breath, fatigue, weight loss, memory loss, bone fracture, joint pains, headache, neurological problem, bleeding, facial swelling, voice change and change of sputum color. Once the patient has been affected by cancer symptoms, different screening procedures (Wang et al. 2015) like genetic testing, scopy bronchi, reflex testing, fluid biopsy, biopsy, and blood testing can be continuously used for evaluation.

The National Institute for Health and Care Research (NIHR) offers general criteria for accurately predicting lung cancer and its stages based on screening technologies. Although it is challenging to maintain prediction accuracy, the screening techniques that have been mentioned successfully analyze lung cells and variations in cells that are used for predicting lung cancer. Among the screening methodologies, computerized tomography (Sfeir, Agnel and Symington 2017) is one of the effective screening processes that accurately examines the deviations and changes present in the human body that are detected by passing X-rays on the human body. Compared to the PET and MRI screening procedure, CT scan - the 30-minute X-ray passes successfully check internal organ function and efficiently collect facts about the damaged portion and tissues. (Shakeel, Burhanuddin, and Desa 2022).

CT scans quickly capture a number of X-ray images. The X-ray images are combined to provide images of the scanned area's soft tissues and bones. Low-dose computed tomography, often known as a low-dose CT scan or LDCT, is the only screening test for lung cancer that is advised by the Centers for Disease Control and Prevention. An X-ray machine employs a low dose

(amount) of radiation to provide detailed images of the patient's lungs while the patient is lying on a table for an LDCT scan.

Figure 2.3 Computed Tomography (CT scan)

CT (computed tomography) imaging is the best imaging technique for diagnosing lung cancer because it can identify all known and unknown nodules (Kareem et al. 2021). Early detection of lung cancer through CT screening can help make the disease more treatable (Venkatesh et al. 2023). An automatic technique for the detection of lung cancer can be developed utilizing CT scans. The system makes use of multiple procedures, including image noise removal, region segmentation, cancer feature extraction, feature selection, and cancer detection. Compared to MRI and PET scans, there is less noise in lung CT images. Consequently, cancer is diagnosed via CT scans. The CT image has the added benefit of being very clear and having minimal distortion and noise. (Venkatesh et al. 2023).

To summarize, CT scans are preferred for lung cancer detection due to their superior sensitivity and spatial resolution, which allow for the visualization of small pulmonary nodules and subtle parenchymal changes that may indicate early malignancy. The tomographic imaging provided by CT enhances the detection of lesions that standard chest X-rays might miss, facilitating earlier diagnosis and improved treatment planning (National Cancer Institute, 2021).

2.2.1 Computed Tomography (CT) Windows

Healthcare practitioner consider computed tomography (CT) scans to be an essential instrument in the field of medical imaging. The idea of CT windows is essential to the interpretation and analysis of CT images since it helps to visualize and assess various tissue types in an efficient manner (Hsieh, 2009). CT windows refers to the display settings that enable doctors and radiologists change the brightness and contrast of CT images so they may highlight particular tissue densities with clarity. The window width and the window level are the two main characteristics that identify these windows. (Bushberg et al., 2012).

The window width determines the range of CT attenuation values that will be displayed within the image, while the window level sets the midpoint of this range. Depending on the particular diagnostic requirements, healthcare providers can optimize the visualization of various anatomical features, such as soft tissues, bones, or air-filled cavities, by altering these parameters (Boas & Fleischmann, 2012). Organs and muscles may be evaluated using a soft tissue window, whereas the skeletal system would be better examined using a bone window. Additionally, specific windows-like the lung window-can improve lung visualization and help identify pulmonary illnesses. (Hsieh, 2009).

Healthcare practitioners who analyze and diagnose patients using medical imaging studies, as well as researchers who work with CT scan images, need to be able to use and comprehend CT windows. By becoming proficient in this method, healthcare professionals can improve patient care and results by improving their capacity to recognize and evaluate a variety of medical disorders.

In our study, we utilized the lung window. When it comes to recognizing and classifying lung cancer, deep learning models can benefit greatly from the insights provided by the lung window, which is specifically made to improve the visualization of the lungs. Deep learning algorithms are better able to learn the characteristic features and patterns linked to various forms of lung cancer by utilizing the best contrast and tissue density information found in the lung window. This enables more accurate diagnosis and early detection of this serious illness (Shen et al. 2015). Using the specialized lung window in conjunction with deep learning's powerful capabilities can be a great combination in the fight against lung cancer, eventually benefiting patients with faster and more accurate diagnoses.

2.3 Deep Learning Algorithm

Machine learning algorithms and Deep learning algorithms are the two emerging techniques that have recently attracted many researchers(Shrestha and Mahmood 2019). In computer vision, the application of deep learning has also shown great promise. These methods reduce users of the burden of difficult handcrafted feature extraction by offering a standard framework for feature extraction. Additionally, a deep learning technique has the potential to enhance the accuracy of early detection of disease. (Latif et al. 2019).

Deep learning plays a significant role in disease detection by leveraging its ability to extract complex patterns and relationships from large and diverse datasets (Shafi et al. 2022). Deep learning models are often trained on enormous volumes of medical data, such as genetic data, clinical notes, medical imaging scans, and electronic health records, in order predict diseases. These models automatically acquire and describe complex properties from the input data by using numerous layers of artificial neural networks. (Cao et al. 2011). By analyzing these learned features, deep learning models can make accurate predictions about various diseases, such as cancer.

The early detection of suspicious nodules can be achieved by performing analysis on computed tomography (CT) scans (Shafi et al. 2022). The top, bottom, and front views, respectively, of the scanned images of the body's internal structure provide details. Comparing this to standard X-rays provides more details about blood arteries, soft tissues, and bones. However, because lung nodules reflect surrounding structures (such as blood vessels), it is challenging to identify early-stage lung cancer with chest CT scans. Thus, the creation of a computer-aided design (CAD) is important in order to identify any suspicious nodules. Because deep learning could reduce the number of scans needed to determine whether a nodule is malignant or benign, it is a promising method for categorizing healthy and malignant cancer nodules. (Heuvelmans et al. 2021). Hence, by adapting a pre-trained deep learning model for medical image detection, researchers can greatly reduce the quantity of labelled data needed for training while still building upon prior knowledge from large datasets. This approach uses characteristics that have already been optimized to identify intricate patterns, which not only speeds up the development of the model but also improves performance (Latif et al. 2019).

2.3.1 Convolutional Neural Network

According to (Alzubaidi et al. 2021), Neural networks are a subset of machine learning, and they are at the heart of deep learning algorithms. They are made up of node layers, which have an output layer, an input layer, and one or more hidden levels. Each node connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network (Rajasekar et al. 2023).

Nowadays, however most of studies are focused on feedforward networks, there are other varieties of neural nets that are applied to many use cases and data types. For example, convolutional neural networks, or CNNs, are more commonly utilized in computer vision and disease detection applications, while recurrent neural networks are more commonly used in speech recognition and natural language processing applications (Rong et al. 2021). Before CNNs, objects in images were identified using challenging, manual feature extraction techniques. Conversely, convolutional neural networks now provide a more scalable way to identify objects in images and classify images by employing the ideas of linear algebra and matrix multiplication to identify patterns in images. However, they can be computationally demanding, requiring GPUs in order to train the models (Alzubaidi et al. 2021).

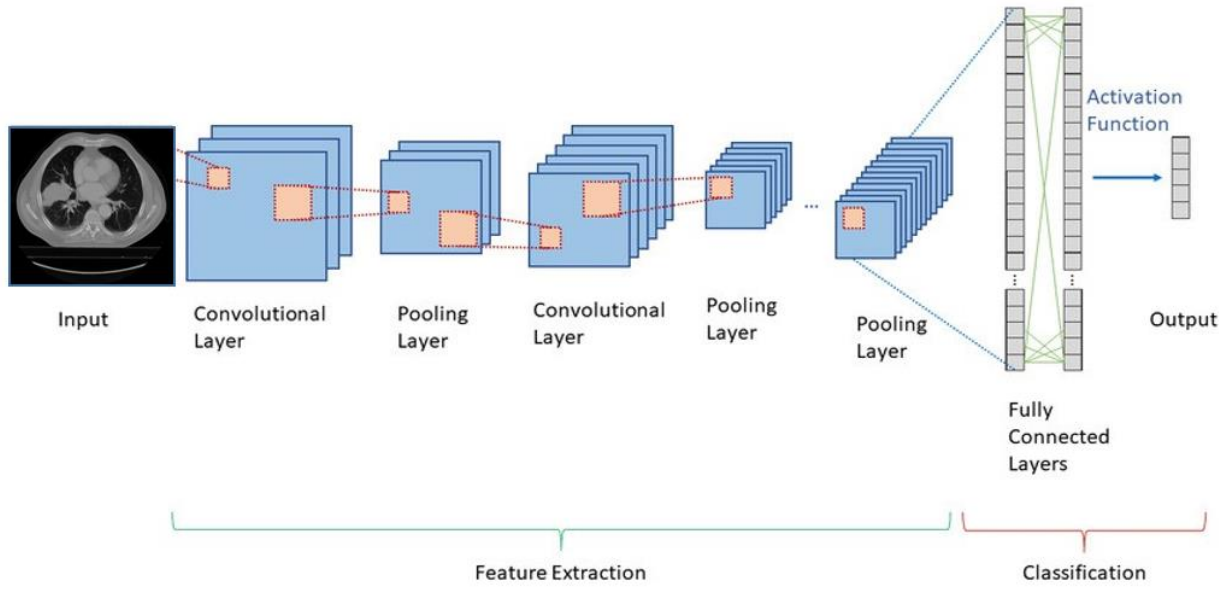


Figure 2.4 General Architecture of Convolutional Neural Network

(Adapted from SLCV, Salman Khan et. Al., 2018)

The science of computer vision has undergone a significant transformation due to the unique architecture of Convolutional Neural Networks (CNNs), which are able to automatically and adaptively learn the spatial hierarchies of information. An input image is first processed through one or more convolutional layers in a standard CNN architecture. Convolutional filters are used by these layers to capture local patterns and features, like edges and textures. Pooling layers are used after each convolutional layer to decrease the spatial dimensions of the feature maps, hence down sampling the data without losing the most important information. By repeatedly adding convolutional and pooling layers, the network is able to build up progressively more abstract representations of the input.

The CNN moves to the fully connected layer, where the high-level features are compressed into a single vector, after the feature extraction process is completed. This layer is essential to classification since it creates predictions by combining the features that have been extracted. To add non-linearity and generate the final output, an activation function - such as Sigmoid or SoftMax—is typically added after the fully connected layer. CNNs are especially useful for applications like imagine identification and classification because of their architecture, which makes it possible for them to learn from huge datasets quickly(Simonyan, Vedaldi, and Zisserman 2014). CNNs are important in modern artificial intelligence because they can perform very well in a variety of vision-related applications by utilizing hierarchical feature learning.

In neural networks, two popular activation functions that have varying applications based on the context of the task are sigmoid and SoftMax. For binary classification issues, the sigmoid function - which returns values between 0 and 1- is very helpful since it can accurately simulate the likelihood of a binary outcome. Nevertheless, it has issues with the vanishing gradient, particularly in deep networks. On the other hand, as it normalizes the output values into a probability distribution over several classes, the SoftMax function is usually used in the output layer of multi-class classification models. This makes sure that the expected probabilities add up to one, making it possible to understand class likelihoods with clarity. Both roles contribute in introducing non-linearity and directing the learning process, but the particular uses for which they are employed show how adaptable neural network topologies are to different categorization problems.

2.4 Transfer Learning

Traditional machine-learning approaches have demonstrably achieved success across various real-world applications, but inherent limitations persist (Dai et al. 2007). A large amount of labelled training data that closely resembles the distribution of test data is necessary for the best results. Nevertheless, obtaining such large labelled datasets can be difficult and frequently requires substantial financial and organizational commitments.

One strategy to mitigate this data scarcity is semi-supervised learning, which leverages a combination of limited labeled data and a larger pool of unlabeled data (Yang et al. 2013). Although the goal of this strategy is to improve learning accuracy by adding unlabeled data, collecting enough unlabeled instances can also be difficult, which eventually reduces the effectiveness of conventionally trained models.

Transfer learning emerges as a promising alternative machine learning technique that tackles these data-related problems by capitalizing on knowledge transfer across domains (Kim et al. 2022). A pre-trained model created for one task is essentially used as an anchor for a new model that deals with a different but related task. This is especially useful when there are significant similarities between the two tasks, or when training for the new task is limited by data. The new model can learn the new task more quickly and effectively by using the pre-trained model's learnt features. It can also prevent overfitting because the model already has a foundation of generalisable features that it can apply to the new task (Shaha 2018).

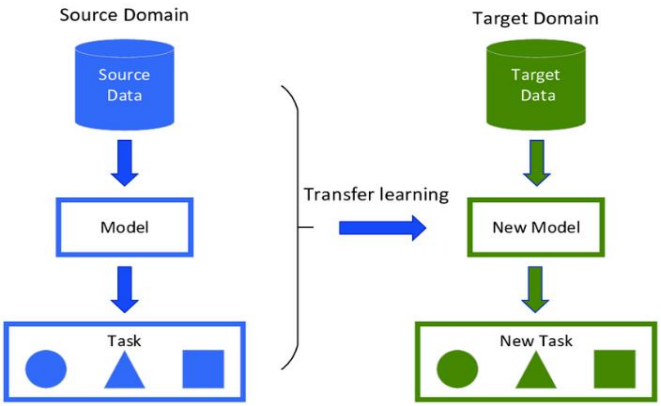


Figure 2.5 Basic layout of the standard transfer learning process.

Adapted from (Algorithm, Li, and Wei 2022)

2.4.1 Transfer Learning with CNN Pre-trained models

We used pre-trained networks in our study on lung cancer diagnosis, using the idea of transfer learning to enhance model performance. In particular, we applied four cutting-edge architectures - DenseNet169, InceptionV3, Xception, and ResNet50 - all of which have become known for their capability in handling challenging medical imaging problems. We will go into much more detail on each model's architecture in this section, emphasizing their unique features and how they improve the accuracy and efficiency of our lung cancer detection tasks.

DenseNet169 is a convolutional neural network (CNN) architecture that is designed to improve feature propagation and reduce the number of parameters required for efficient learning. Dense connection, in which every layer gets input from every layer before it and transfers its output to every layer after it, is what distinguishes its distinct architecture. This method effectively addresses the vanishing gradient issue that frequently happens to deep architectures by facilitating improved gradient flow during backpropagation (Huang et al. 2017). The combination of batch normalization and ReLU activation functions in each layer significantly enhances the model's capacity to learn complicated features from medical imaging. With 169 layers, including a number of dense blocks and transition layers that effectively compress feature maps, DenseNet169 is especially skilled at processing high-dimensional input (Xie, Girshick, and Doll 2017).

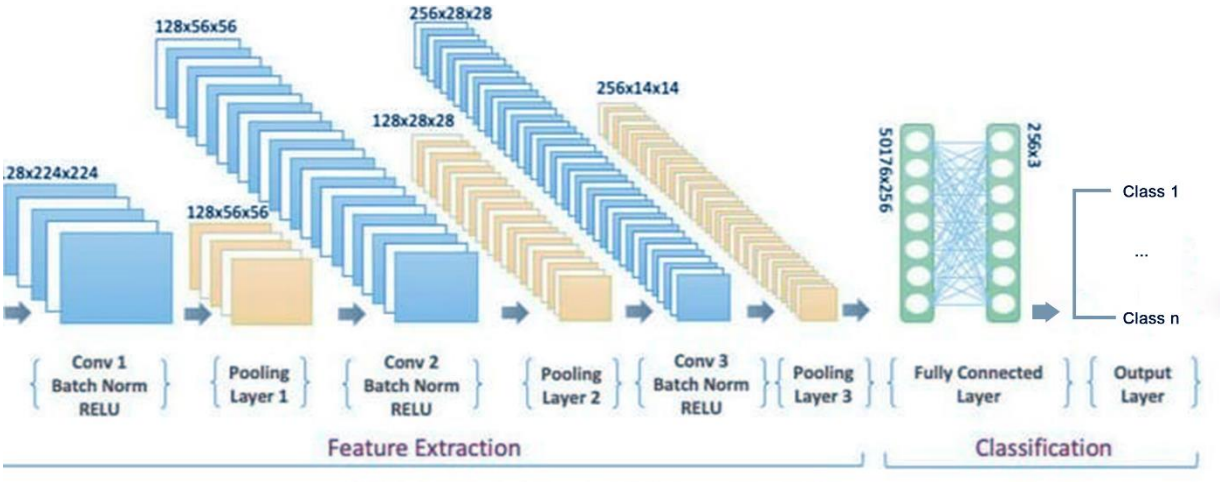


Figure 2.6 Architecture of DenseNet169

Adapted from (Mali Patil et al. 2023)

The standard size for input images required by this model is 224x224 pixels, which arises a balance between computing efficiency and detail. Because the architecture is built to handle RGB color images, it can be used for lung scans that make use of color-enhanced imaging methods, such computed tomography (CT) scans (Huang et al. 2017). The network's extensive connectivity across layers facilitates the effective propagation of characteristics, which improves the model's capacity to identify complex patterns in the data.

InceptionV3 is especially useful for detection tasks because it maximizes both accuracy and processing efficiency with its very advanced convolutional neural network (CNN) architecture. It can handle RGB color images, which are typical in medical imaging modalities like computed tomography (CT) scans, and requires input images with a pixel size of 299x299. The use of inception modules, which enable the network to do convolutions with several filter sizes simultaneously and collect information at different scales, is one of InceptionV3's unique characteristics (Szegedy et al. 2017). This multi-scale processing is critical in recognizing minor differences in images that may suggest target finding. InceptionV3 additionally includes batch normalization and auxiliary classifiers to increase gradient flow and training stability. Research has demonstrated that InceptionV3 works better than a lot of conventional CNN designs in medical imaging tasks, such as detecting cancer, by obtaining a higher diagnostic accuracy while keeping the number of parameters under control (Mali Patil et al. 2023).

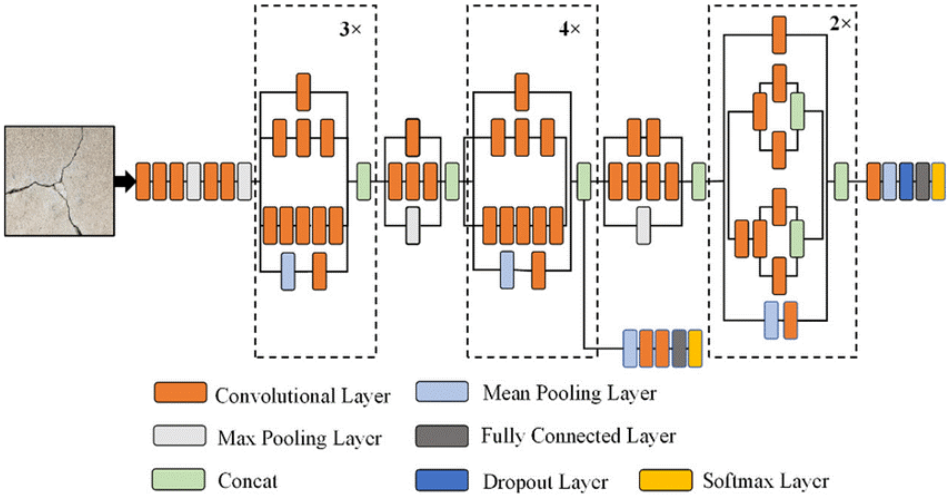


Figure 2.7 Architecture of InceptionV3

Adapted from (Ali et al. 2021)

A state-of-the-art convolutional neural network (CNN) architecture called Xception improves performance in a range of image classification applications. It can handle RGB color images and usually requires input images that are 299 by 299 pixels in size, which makes it adaptable to a variety of uses. Effective feature extraction is made possible by the architecture's use of depth-wise separable convolutions, which dramatically reduce the number of parameters while preserving high representational capacity (Chollet 2017). Xception can easily catch complicated details and elaborate patterns in images because of to this design. Extensive research has demonstrated that Xception surpasses conventional CNNs on several benchmarks, exhibiting improved precision and effectiveness in assignments like item identification and scene categorization (Rajpurkar et al. 2017). Its ability to learn detailed image representations positions Xception as a valuable tool in the field of deep learning and computer vision.

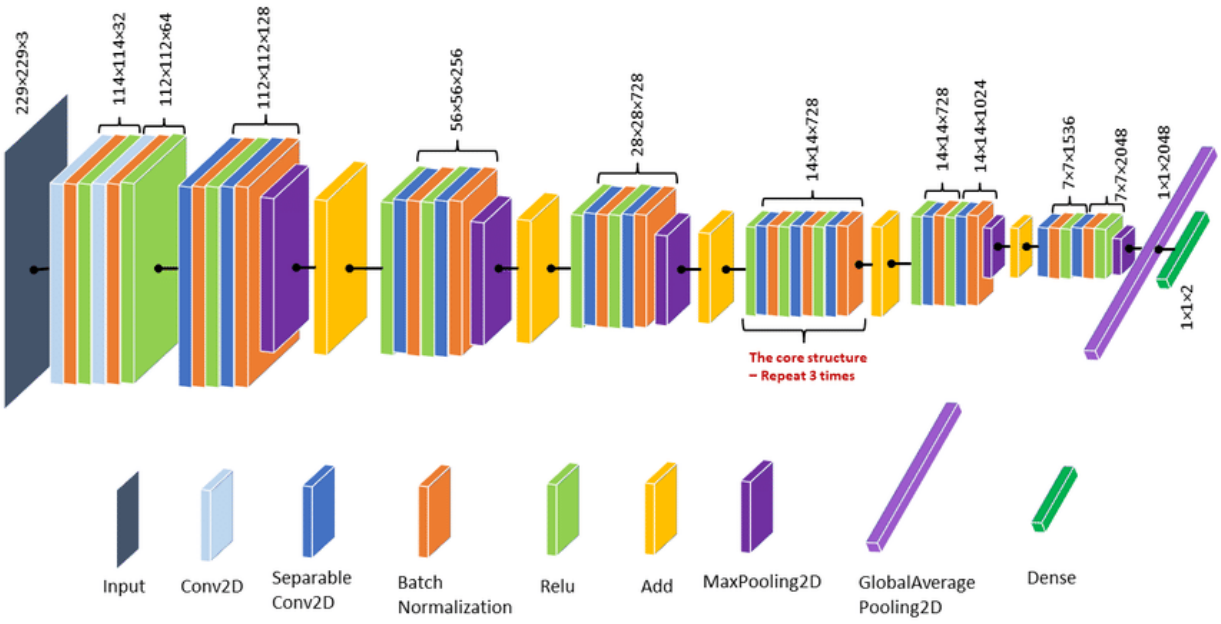


Figure 2.8 Architecture of Xception

Adapted from (Mehmood 2021)

ResNet50 is a convolutional neural network (CNN) architecture with significant influence renowned for its capacity to efficiently train extremely deep networks. It can process RGB color images and takes 224 × 224-pixel input images, which makes it appropriate for a variety of image classification jobs. ResNet50's use of residual connections is one of its distinguishing features. These connections enable the model to learn identity mappings and address the vanishing gradient issue, making it possible to train networks with a lot more layers than in conventional architectures (He et al. 2016). ResNet50's outstanding performance levels are made possible by its creative design, which also preserves computing complexity. To add to its robustness, the 50-layer architecture combines batch normalization, ReLU activations, and convolutional layers. ResNet50 is a fundamental model in deep learning and computer vision because of its exceptional performance on a variety of benchmarks, such as object detection and image categorization (He et al. 2016).

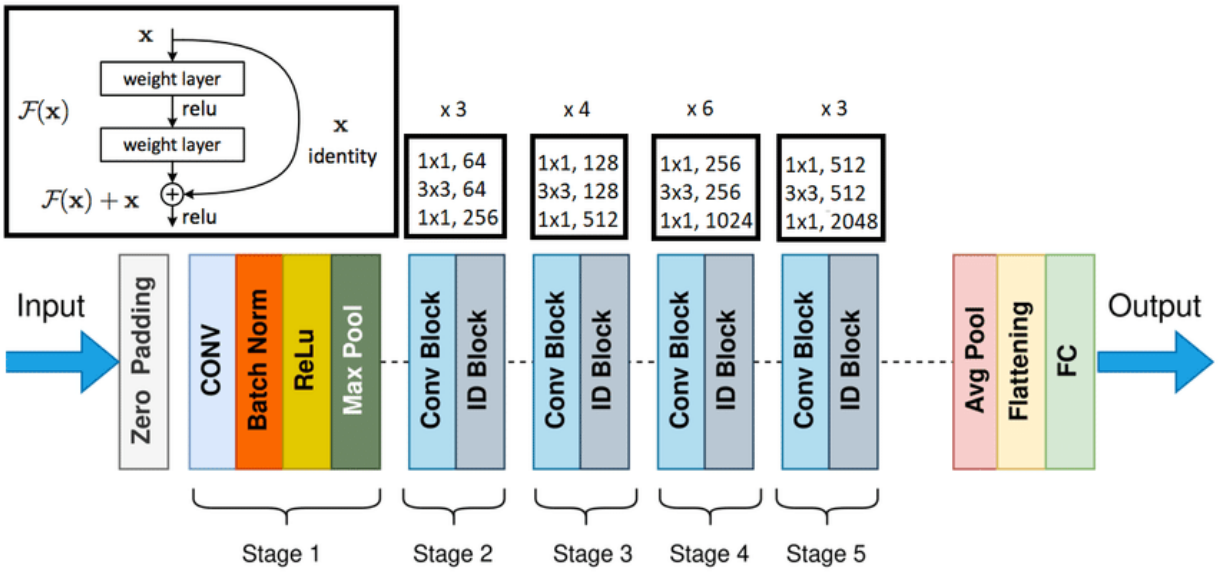


Figure 2.9 Architecture of ResNet50

Adapted from (Gomes et al. 2022)

2.5 Explainable Artificial Intelligence (XAI)

To mitigate the black-hole nature of deep neural network and enhance a trust of its detection we employed Explainable Artificial Intelligence techniques. Many layers joined by numerous nonlinear entangled interactions make up a typical neural network. Understanding how the neural network arrived at its conclusion is not feasible, even if one were to examine each of these layers and explain how they relate to one another. Consequently, deep learning is frequently referred to as a 'black box'. In several application domains, there is growing concern that these 'black boxes' could be biased in some way, and that this bias may go unreported. This may have far-reaching effects, particularly in medical applications. (van der Velden et al. 2022)

Approaches to comprehend the black box better have been called for. These methods are often called explainable artificial intelligence (XAI) or interpretable deep learning. Since both words are frequently used interchangeably, we'll go with XAI. The United States Defense Advanced Research Projects Agency (Krotkov and Blitch 1999) and the Association for Computing Machinery conferences on Fairness, Accountability, and Transparency are two prominent examples of XAI activities. XAI is being used more and more by medical imaging researchers to interpret the output of their algorithms. If something clarifies how a neural network arrived at a choice or makes the decision understandable, it might be deemed a good explanation.

2.5.1 GradCAM

GradCAM - Gradient-weighted Class Activation Mapping - is one of backpropagation-based visualization techniques. Grad-CAM is a generalization of CAM that was introduced by (Selvaraju et al. 2019). Grad-CAM is capable of producing post hoc local explanations with any kind of CNN, but CAM requires global average pooling particularly. Additionally, guided Grad-CAM, an element-wise multiplication of Grad-CAM and guided backpropagation was presented by the authors. Medical image analysis has made use of Grad-CAM and Guided Grad-CAM. GRAD-CAM was used, for instance, (Ji 2019) to demonstrate which histology lymph node sections a classifier relied upon when determining the presence of metastatic tissue; by (Kowsari et al. 2020) to identify small bowel enteropathies on histology; and by (Windisch et al. 2020)to demonstrate which brain MRI regions the classifier relied upon when determining the presence of a tumor. Several studies have effectively utilized Grad-CAM for medical image visualization,

demonstrating its value in enhancing transparency. (Panwar et al. 2020) developed a deep learning and Grad-CAM-based color visualization approach for the rapid detection of COVID-19 cases using chest X-ray and CT scan images. This method allowed clinicians to easily identify the regions of interest that contributed to the model's predictions, significantly improving diagnostic speed and accuracy. In the work on artificial intelligence-based pathological examination of liver cancer, (Yan et al. 2023) demonstrated how Grad-CAM may be used to interpret model predictions in histopathology images, offering important information about the characteristics of tumors and enhancing the precision of diagnosis. These applications highlight how Grad-CAM enhances the credibility of deep learning in medical situations while also helping to understand model predictions.

2.5.2 LIME

LIME - Local Interpretable Model-Agnostic Explanations - is a widely used technique for providing explanations of machine learning model predictions. Introduced by (Ribeiro, Singh, and Guestrin 2016), LIME generates local approximations of the decision boundary by perturbing the input data and observing the resulting changes in predictions. This method allows for interpretable explanations of any classifier, including deep neural networks, by approximating the model's behavior in the vicinity of a particular prediction.

In the context of medical image analysis, LIME has been effectively employed to elucidate the reasoning behind AI-based diagnoses. For instance, (Ghafoor et al. 2020) utilized LIME to interpret predictions made by models analyzing chest X-rays, helping radiologists understand which features were most influential in the AI's decision-making process. Similarly, (LaLonde et al. 2021) applied LIME to skin lesion classification, revealing critical features within images that contributed to the model's outputs.

The use of LIME enhances the interpretability of complex models and fosters trust among medical professionals by providing clear insights into the factors driving predictions. This is particularly important in clinical settings, where understanding the rationale behind a diagnosis can influence treatment decisions. By integrating LIME into medical image analysis workflows, clinicians can better validate and trust AI-assisted diagnoses, ultimately leading to improved patient care and outcomes.

2.6 Related works

Cancer is a common disease with an increasing mortality rate in recent years. Lung cancer is the most common cancer in men and women alike (Venkatesh et al. 2023). So, many works have already been proposed for the detection and classification of lung cancer by various researchers using different deep learning, machine learning, and transfer learning models. Each study has been summarized and explained as below.

(Thi et al. 2024) proposed CNN architectures with DenseNet121, MobileNetV2, InceptionV3, InceptionResNetV2, ResNet50, ResNet101, VGG16, VGG19, and Xception for the classification of lung cancer and pneumonia in 2024. They used CT lung Pneumonia as non-cancer from peer-reviewed publications and Lung CT data from CIA (Cancer imaging archive). Their main goal was to integrate Transparency to their model's decision. It is one of the paper successes in incorporating the idea of XAI based Transparency. They used GradCAM as a visualization technique. They reported that, ResNet50 demonstrated the highest accuracy and sensitivity (97.7% and 100%, respectively), while InceptionV3 excelled in precision (97.9%) and specificity (98.0%). They reported that they used Adobe Photoshop to eliminate a caption from DICOM image which is tedious job and untrusted image manipulation. Also, considering only a pneumonia image as non-cancer is scientifically not competitive. In our case we considered non-cancer image is an image with all other lung disease, and also health lung while lung cancer covers primary tumor and metastasis. Also, each class's image has different value of image quality and size, for they were collected at different time and for different purposes.

(Shakeel, Burhanuddin, and Desa 2022) proposed a model that detect the presence or the absence of Lung cancer using CT scan images. They gathered a dataset from the cancer imaging archive (CIA) and the dataset consists of 5043 CT images of which 3500 images are used for training, and the remaining 1543 images are used for validation. The study optimizes image processing techniques to enhance the quality of lung images and utilizes a hybrid intelligent spiral optimization-based generalized rough set approach to select optimized features. The selected features are then classified using an ensemble classifier, resulting in improved accuracy in lung cancer prediction compared to traditional techniques. From the proposed model's Ensemble classifier using the Genetic algorithm with wrapper approach (GAWA) feature selection method outperformed with an Accuracy of 97.6%. One potential drawback of the paper

is that it lacks a comparison with other state-of-the-art lung cancer detection systems. While the paper does compare its results to other feature selection and classification algorithms, it does not compare its overall performance to other existing systems like transfer learning with pretrained CNN models. Also, however the accuracy outcome of the model seems high, the transparency of the model is not implemented for the clinical trustworthy.

In 2022 (Shafi et al. 2022) deployed a deep learning-assisted SVM-based model that yields 94% accuracy for pulmonary nodule detection representing early-stage lung cancer. That means a Convolutional Neural Network is used for feature selection while SVM is utilized to classify the cancer. For their study, they used the LUNA16 dataset and investigated 888 annotated CT scans from the publicly available LIDC/IDRI (Lung Image Database Consortium (LIDC) and Image Database Resource Initiative) database. The author demonstrated that their model was found superior to other existing methods including complex deep learning, simple machine learning, and the hybrid techniques used on lung CT images for nodule detection. In their study, the max pooling capsule layer - which is used to detect features—was replaced with a capsule network (CapsNet) in order to get over CNN's limitations. Size, position, and hue are identified by a single capsule (a group of neurons) in the capsule layer. The length of the vector, which establishes the likelihood that features would be present in the input, is the capsule's output. The vector's orientation is utilized to qualify the capsule attribute. Because capsules are autonomous, there is a greater chance of finding the desired conclusion when several capsules agree.

(Kapileswar et al. 2022) proposed a hybrid CNN-LSTM model used for Lung cancer detection using CT scans. While their primary aim is to improve the accuracy of the model in lung cancer detection, their attention was improving the image quality for better model training. They used a CT scan of 1010 patients from the LIDC-IDRI database with a total of 244,527 DICOM images. Finally, as per their proposal, their model provided an accuracy of 97%. However, their model performed at the higher accuracy, they didn’t employ the concept of transparency visualization, and image segmentation is the only step they took to improve the attention.

In 2023 (Venkatesh et al. 2023) proposed a Patch Processing & CNN model that performed an accuracy of 99.6% in the detection of lung tumors in CT images. They used a public dataset of CT images from the Lung Image Database Consortium (LIDC). According to their study, CNN is used in this work for classification and feature extraction. In their study, they divided the

image into small patches and processed each patch separately instead of focusing on the entire image. Patching is an appropriate technique for removing undesirable flaws from an image because it allows you to replace selected shaped portions with a surface suited to other arbitrarily shaped regions and a synthetic noise component. (Venkatesh et al. 2023) could achieve an accuracy of 99.6% in detection of lung cancer, but the clinical trust of their model is not addressed for they didn't considered the transparency of the result behind their classifier.

In the study of (Kareem et al. 2021) support vector machine (SVM) is used to classify lung cancer as Normal, Benign, or Malignant. To train their model they used 1100 CT images from IQ-OTH/NCCD lung cancer dataset. Their best accuracy is 89.8876% by evaluating different SVM kernels and feature extraction techniques. This result is achieved when using SVM with a polynomial kernel and a combination of two extracted features, Gabor filter and GLCM. It is straightforward to evaluate their model performance from their accuracy which is low compared with other reviewed papers. So, more probably it is because of their preprocessing techniques applied to the image dataset. Also, they are supposed to try an experiment with state-of-the-art algorithm, especially pretrained CNN models, with the concept of transfer learning.

In (AL-Huseiny and Sajit 2021) work transfer learning is used to readjust GoogLeNet deep neural network to learn medical data. This includes allowing the final layers of the DNN to evolve while restricting deep layers. In this setting, a rough, unprocessed dataset, the IQ-OTH/NCCD lung cancer dataset was used to train/validate the proposed algorithm. According to their experimental results, this algorithm scores 94.38% accuracy, which outperforms the benchmark method previously used with this dataset according to their claim. According to their proposal, deep neural networks (DNN) are selected to detect images with malignant nodules of lung computed tomography (CT). The method includes subjecting input images to a simple and fast pre-processing that isolates regions of interest (ROI), that is the lungs-dominated area, ridding the images of other surrounding tissues and artifacts. Centered and size-normalized images are then fed to a deep neural network for training and validation. But they should have an experiment with other models to compare the result with GoogleNet plus models Transparency approach is missed.

In 2022 (Humayun et al. 2022) proposed a transfer learning approach with CNN to classify lung carcinoma. In their study, they used three preprocessed models VGG16, VGG19, and Xception which resulted in an accuracy of 98.83%, 98.5%, and 97.4% respectively at the 20th epoch. They used a CT scan image dataset from the Iraq-Oncology Teaching Hospital/National Center for Cancer Diseases (IQ-OTH/NCCD) for their study. Accordingly, their dataset contains three classes namely Normal, malignant, and Benign from 1190 images where 55 cases were classified as normal, 15 cases were classified as benign, and 40 cases as malignant. Their study specifies how much adapting a preprocess model is feasible in terms of resources, performance and time. But they utilized very small dataset size relative to another lung cancer-based study.

Table 1 - Summary of Related Works

Researcher	**Title**	**Contribution**	**Research Gap Observed**
Thi et al. 2024	The application of deep learning in lung cancerous lesion detection	They integrated model transparency, also they tested with nine pretrained CNN models.	• Lack of image generality, Untrusted image manipulation.
(Shakeel, Burhanuddin, and Desa 2022)	An Automatic lung cancer detection from CT image using improved deep neural network & ensemble classifier	They employed multilevel brightness-preserving approach to enhance the quality of lung images	• Lacks of comparison with other state-of-the-art lung cancer detection systems • No transfer learning
(Shafi et al. 2022)	An Effective Method for Lung Cancer Diagnosis from CT Scan Using Deep	Employed capsule network (CapsNet) for feature extraction to	• Lack of extended experiment with CNN architectures

	Learning-Based Support Vector Network	overcome the limitation of CNN	
(Kapileswar et al. 2022)	Deep Learning Algorithm for Classification and Prediction of Lung Cancer using CT Scan Images	To increase the model's accuracy, they enhanced the method for enhancing image quality.	• Lack of extended experiment with CNN architectures
(Venkatesh et al. 2023)	A hybrid model for lung cancer prediction using patch processing and deep learning on CT images	They incorporated the concept of patch processing to remove an undesirable flaws from an image	• No transfer learning
(Kareem et al. 2021)	Evaluation of SVM performance in the detection of lung cancer in marked CT scan dataset	Numerous SVM kernels and feature extraction techniques are evaluated.	• Low model Performance • No transfer learning
(AL-Huseiny and Sajit 2021)	Transfer learning with GoogLeNet for detection of lung cancer	They exposed the power of GoogleNet DNN in learning medical data as a transfer learning	• Much lower accuracy • Experiment limited to GoogleNet only •

(Humayun et al. 2022)	A Transfer Learning Approach with a Convolutional Neural Network for the Classification of Lung Carcinoma	The proposed model utilizes small computational complexity related to other studies.	• Too much small Dataset size

All of the above-related works provide valuable input for our study; however, the paper by (Thi et al. 2024) stands out as a key paper in terms of its similarity with our study, due to its innovative application of transfer learning combined with Grad-CAM for visualizing regions of interest in lung CT scans. This method makes it possible to grasp model predictions more easily, which is important when using medical contexts. However, their dataset has several significant shortcomings, as discussed in Section 2.6. In particular, it mainly concentrates on pneumonia images classified as non-malignant and cancerous lung CT images labelled as cancer. The inclusion of normal lung images and another lung disease, which are necessary to create a full model that can differentiate between normal and cancerous diseases, is overlooked by this binary categorization. Moreover, focusing only on one CT window - the lung window, in this case - restricts the study and could lead to bias since it ignores the variability found in other CT window settings. This constraint raises are uncertain on the validity and applicability of their conclusions, highlighting the need for a more diversified dataset that includes all possible lung CT scan appearances in order to improve diagnostic precision in clinical settings.

Chapter Three

3. Materials and methods

3.1 Chapter Overview

This chapter presents the methodology that was used to carry out this study. It provides an overview of the procedures followed to develop and evaluate lung cancer detection model. It also discusses data collection and data augmentation techniques utilized to expand the dataset. Finally, it presents evaluation metrics and, the software libraries and hardware utilized for model training and development.

The research process followed multiple stages to achieve the goal of developing lung cancer detection model. The process employed in carrying out this study involves identifying the research area, reviewing previous research by different researchers, investigating limitations in previous works, formulating research questions, selecting appropriate tools and methodologies, collecting and preparing datasets, designing and implementing proposed solution and finally writing a report for thesis. Figure 3.1 shows the high-level overview of research design process from identifying research area to model development, model evaluation and report writing.

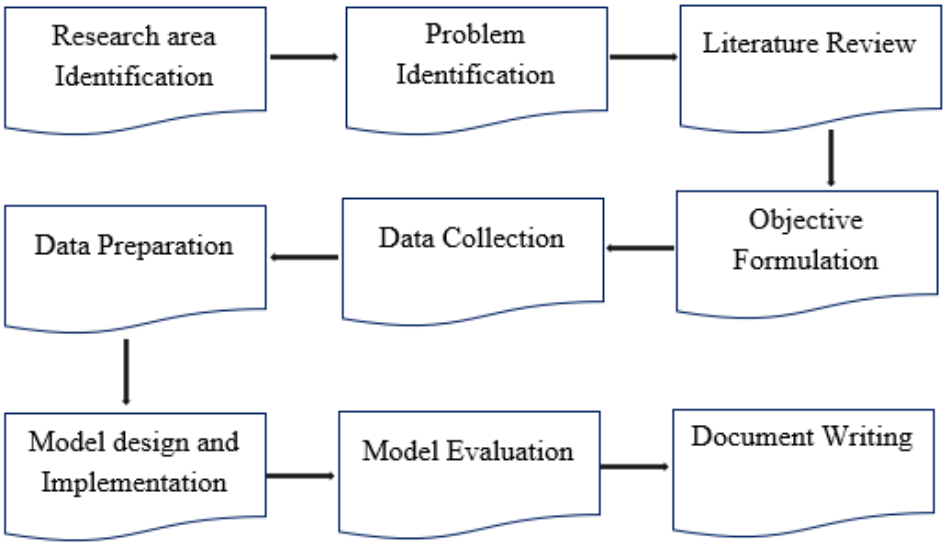


Figure 3.1 Overall Research Process

3.2 Data Collection

Data is mandatory to develop the model in a deep learning algorithm. Without data, there is nothing to do with deep learning algorithms. The size of the data required depends on the problem that will addressed (Lecun, Bengio, and Hinton 2015). To develop a lung cancer detection model, the primary task was collecting CT image based medical records of the patients. Lung cancer is a complex disease with various subtypes and stages, and there can be significant variability in imaging data, such as CT scans (Van der Straeten and Tasson 1985). To capture this variability and ensure the model's robustness, it is important to have a diverse dataset that covers different subtypes, stages, and variations in imaging techniques. Including data from multiple sources or collaborating with multiple institutions can help increase the diversity of the dataset (Shrestha and Mahmood 2019).

According to (Humayun et al. 2022), in Ethiopia, there is no nationwide cancer registry and there is only one oncology referral hospital located in Tikur Anbessa Specialized Hospital (TASH), Addis Ababa, Ethiopia. So, the dataset for our study was collected from TASH and annotated with the help of radiologist experts to label the CT scan images to their correct classes as normal lung or cancerous lung. According (Gebremariam et al. 2021) radiologists use CT scan features like Lung mass, Pleural effusion, Lung nodules, Metastasis to other sites, Collapse consolidation, Lymphadenopathy, Lung opacity, Lymphangitic carcinomatosis, Pleural thickening, Erosion of rib, Cavity, Calcification and Pancoast tumor to classify lung cancer as Normal, Malignant or Benign.

For us to access and gather our target data on their server using their MedWeb software, Tikur Anbessa hospital set up an account for us. Image data in many modalities is accessible on the MedWeb. Once each patient profile was opened, we gathered information using the "CT" modality, "Read" status, and "Chest" type to determine whether or not the patient's lung had cancer. Thus, each patient's profile is downloaded and saved in a separate folder. Afterward, we trimmed and saved each CT slice according to its class - ‘Normal’ or ‘Cancerous’ - with the assistance of a radiologist. 200 patients' total of data were gathered from MedWeb; 60 of them had a lung cancer issue, and 140 were normal. Finally, our dataset contains 2000 normal lung CT and 1500 cancer lung CT, from a total of 3500 CT images after data cleaning is applied.

As we discussed under sub-topic 2.1.1 CT scan image can be accessed and captured through different window, with different contrast and different goal. As of our study we strongly focused lung windows, and included different window images like ‘default’ window for data variability and our models general use. This helps our model to classify a CT image with different windows.

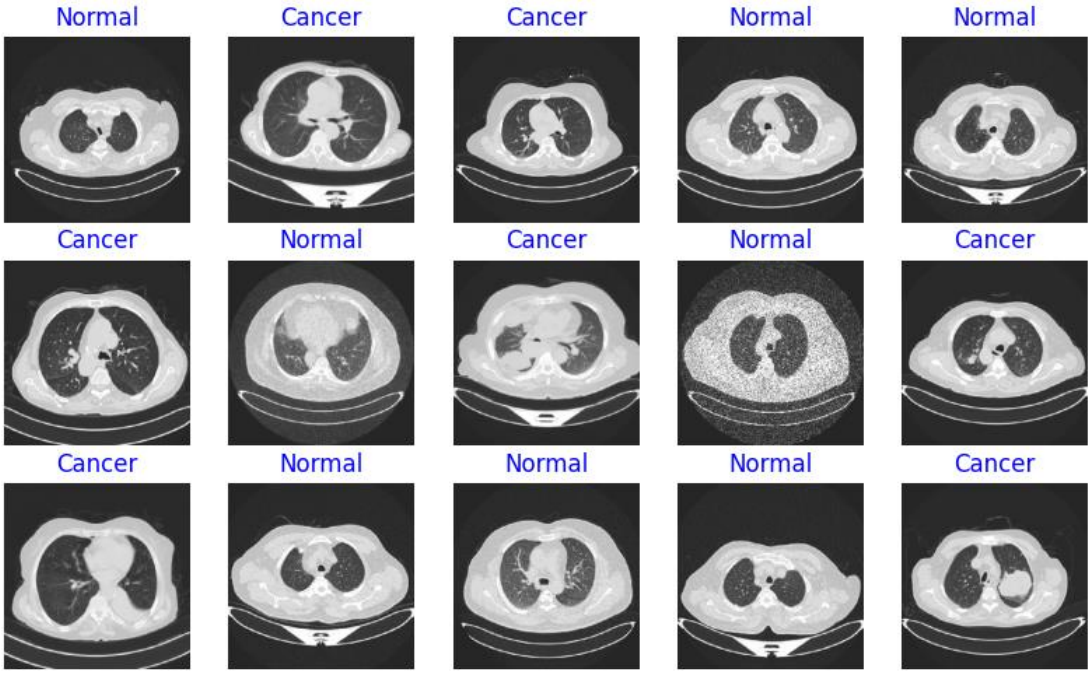


Figure 3.2 Dataset Samples

3.3 Dataset Partitioning

An on-the-fly data partitioning strategy was used for the purpose of lung cancer detection. This choice was motivated by the inherently diversity and complexity of medical imaging data, which can show wide variations in parameters including patient demographics, tumor characteristics, and image quality. During training and validation, we made sure that our deep learning model was exposed to a representative sample of the entire data distribution by employing an on-the-fly partitioning technique. The primary benefit of this approach was its capacity to accurately represent the heterogeneity present in the lung cancer dataset, hence enhancing the model's generalization capabilities. We were able to reduce the possibility of overfitting and make sure the model was not biased towards any particular subset of the data by arbitrarily shuffling and partitioning the data. Hence, we classified our training data and validation data with a ratio of 90%:10% for both the cancer and Normal classes as of our study's data partitioning approach.

3.4 Data Pre-processing

The primary goal of image preprocessing is to convert collected data into suitable data quality before performing any experimental work, particularly in deep learning (Latif et al. 2019). Since these techniques differ depending on the purpose and type of data, the types of preprocessing techniques used in various studies vary. The quality of the images in this study varies, which may have a significant impact on the model's performance. As a result, various preprocessing methods are used. Each of them is revealed step by step in the subsections that follow.

3.4.1 Image Cleaning

Images of low quality that were either too low quality to be classified or did not contain enough lung CT information were removed from the dataset. For the lung cancer detection model to remain accurate and dependable, image cleaning was essential. Higher performance and better generalization resulted from the model's ability to focus on learning from the most representative and informative data by eliminating irrelevant or low-quality images. This step also avoided the possibility of experiencing bias or noise to the training process, which would have had a negative effect on the model's ability to correctly identify and categories cases of lung cancer. The careful curation of the dataset through image cleaning was a vital component of the overall methodology, contributing to the robustness and clinical utility of the final deep learning solution.

3.4.2 Resizing Images

Image resizing is an important preprocessing step in deep learning tasks involving image data, as deep learning models often have specific input size requirements. For the lung cancer detection task, we opted to dynamically adjust the image size in code to match the input requirements of each CNN architectures. By integrating the image resizing into the data preprocessing pipeline, we were able to seamlessly adapt to the varying input size requirements of different deep learning architectures, such as DenseNet (224x224), InceptionV3, Xception, and ResNet50 (299x299). This automated resizing process improved the efficiency and scalability of the workflow, while maintaining the integrity of the original image data.

3.5 Feature extraction

After preprocessing the CT scan image, CNN based pre-trained feature extractor algorithms are employed for feature extraction. These models are capable of extracting meaningful and discriminative information from images since they have been trained on large datasets. We take advantage of the deep architectures of these trained models to extract high-level representations of the input CT scan image. By utilizing various models to extract features from our data, we were able to get complementary and varied information that improved the overall representation of the tumor region. This approach makes use of the transferability of the information acquired by these models to reduce computational resources and improve the accuracy of tasks that are proposed.

So, in our work four pretrained models are deployed for feature extraction. Each model is carefully selected after revising related works and understanding their performance. These models are denseNet169, InceptionV3, Xception and resNet50. These models are trained on ImageNet, which is a huge dataset with so many classes, and different features. These models learned at a lot of layers to extract features from a pioneer dataset, and while their layers are fine-tuned to our work, they performed at a promising accuracy.

3.6 Model

In order to detect lung cancer and more accurately classify unknown data as normal or cancerous, we developed a deep learning algorithm. We employed a Convolutional Neural Network architecture that had been extensively trained on data for this purpose. In order to detect lung cancer, we used CNN architectures. In order to choose the architecture that provides the best performance, we reviewed literatures of related works and selected some cutting-edge CNN architectures that have been trained on enormous datasets - DenseNe169, InceptionV3, Xception and ResNet50. The classifier layer receives the intermediate output of the layer once the feature has been retrieved using these pretrained models. Hence, to reduce the complexity of the huge models we freeze some layers and fine-tune more promising layers on top of the frozen layer. Finally, GradCAM is used to visualize an important part for an image that led to prediction.

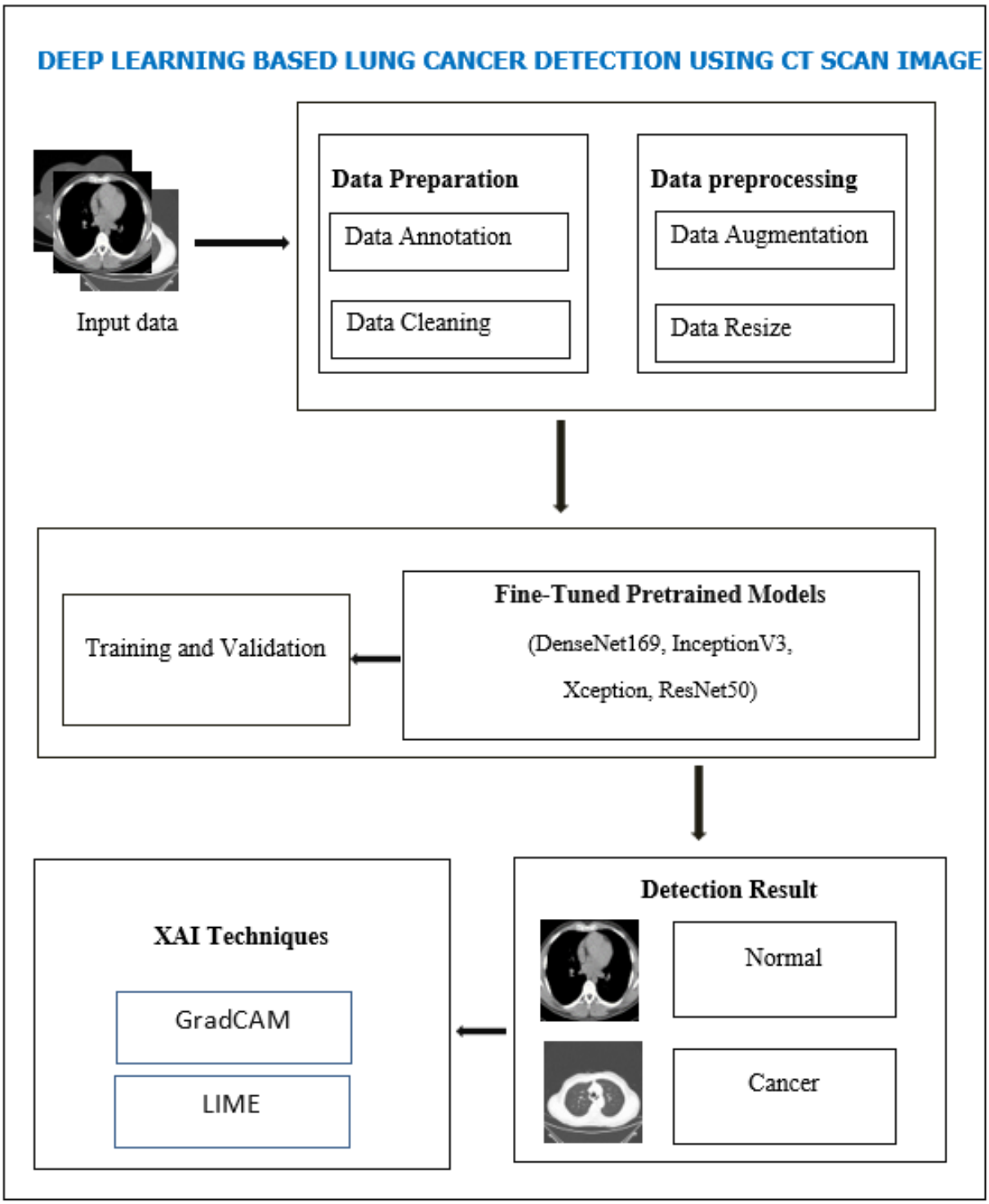


Figure 3.3 Overall process of the proposed solution

3.7 Model Evaluation Metrics

Performance evaluation metrics are used to understand how well the proposed model performed on the input data that was provided to it. They help us to make sure that the model performs correctly and optimally. In this study, we used evaluation metrics such as accuracy, precision, recall, and F1 Score. Accuracy measures the overall correctness of the predictions made by the model. It is the ratio of the number of correct predictions to the total number of predictions.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad \ldots\ldots \quad (1)$$

Precision measures the proportion of correctly predicted positive cases out of the total predicted positive cases. In the context of lung cancer prediction, precision indicates the ability of the model to correctly identify the presence of lung cancer.

$$Precision = \frac{TP}{TP+FP} \quad \ldots\ldots \quad (2)$$

Recall measures the proportion of correctly predicted positive cases out of the actual positive cases. It indicates the model's ability to identify all instances of lung cancer in the dataset.

$$Recall = \frac{TP}{TP+FN} \quad \ldots\ldots \quad (3)$$

The F1 Score is the harmonic mean of precision and recall. It provides a balanced measure of the model's performance by considering both precision and recall.

$$F1\ Score = 2\ x \frac{(Precision\ x\ Recall)}{(Precision+Recall)} \quad \ldots \quad (4)$$ (From 2 and 3)

Another evaluation metric used to evaluate a classification task's performance is a confusion matrix. A confusion matrix, in which TP stands for True Positive, FP for False Positive, FN for False Negative, and FP for False Positive, can be used to compute Precision, Recall, and Accuracy. True Negatives (TN) are accurately recognized negative instances, whereas True Positives (TP) are correctly identified positive instances. Positive cases that are mistakenly classified as negative are known as false negatives (FNs), and negative cases that are mistakenly classified as positive are known as false positives (FPs).

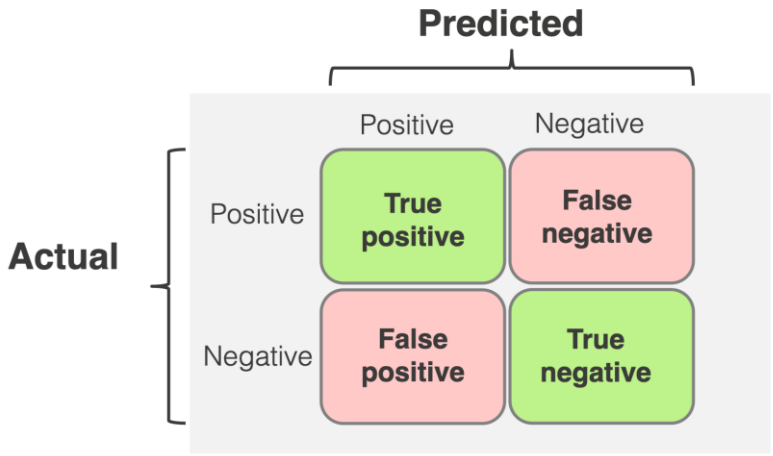


Figure 3.4 Confusion Matrix Terminology

3.8 Overfitting Problem

Overfitting is the main concern in deep learning tasks. It happens in a deep learning model when it is supplied insufficient training data; hence, the model does not generalize new datasets and only uses the data to memories the dependent features. The main reasons for overfitting include the model's complexity, the training set's limited size, and data noise. To prevent the overfitting problem, different regularization procedures are used to adapt the model according to the specific problem scenario.

In order to avoid overfitting, we applied four primary techniques in our study. The first is reducing model complexity by freezing certain layers so that only some layers can learn features that are generalizable to the entire model. The second technique is early stopping, which is pausing training prior to a decrease in performance. The third is regularization, which use strategies like L1 and L2 regularization to remove irrelevant characteristics and choose pertinent ones, so preventing training data noise from being memorized. Last but not least, dropout prevents the network from growing overly dependent on any one feature by arbitrarily disabling a portion of neurons during training. This helps to further encourage generalization.

3.9 Development Tools

For the successful completion of this lung cancer detection study, the we used a comprehensive suite of hardware and software tools. The models were implemented and trained using popular machine learning framework, a TensorFlow, on GPU-accelerated hardware to improve the training process and explore more complex model architectures. By employing this comprehensive suite of tools and materials, we were able to develop and validate deep learning-based solutions for accurate and reliable lung cancer detection, paving the way for successful integration into clinical practice.

3.9.1 Designing Tools

The development, exposition, and interpretation of design concepts will be carried out with these conventional tools. To design the suggested system, Enterprise Architect have been employed. For making network diagrams, flowchart diagrams, and other graphic design tools that look professional, this lightweight and strong tool is ideal.

3.9.2 Hardware Tools

Table 2 - Hardware tools required

No	Tools	Specification	Description
1	Hard Drive	WD 1TB USB 3.2 Gen 1	CT based personal record on a patient we collected from TASH requires large storage for each of them consumes more than 800MB.
2	RAM	16 GB RAM	Training doesn't only need enough size of GPU, also enough size of RAM is recommended for better experience.
3	Flash Disk	16 GB	Flash disk is required to transfer medium sized files among each PCs.
4	GPU	NVIDIA GeForce RTX 2070	GPU is required to perform parallel computations efficiently, speeding up the processing of our dataset.

3.9.3 Software Tools

Table 3 - Software tools required

No	Software Tools	Descriptions
1	Google Colab	A free online resource that offers Jupyter Notebook access and computational power for tasks involving coding, data analysis, and machine learning
2	Jupyter Notebook	open-source mathematical framework that is kept up to date by participants in Project Jupiter. It is used to create and run code in order to carry out the proposed tasks, and it is integrated with an Anaconda environment.
3	Anaconda	It is an application for installing the most recent version of Python together with all of its modules and IDEs and for putting the proposed solutions into practice.
4	TensorFlow	It is an end to end open source, flexible ecosystem of machine learning platform tools.
5	Keras	It is an open source library that provides a Python interface for neural networks and serves as a bridge between Python and TensorFlow.
6	RadiAnt DICOM Viewer	It is a powerful and user-friendly medical imaging software that enables healthcare professionals to view, analyze, and interpret DICOM images efficiently, enhancing diagnostic capabilities and workflow.

Chapter Four

4. Proposed Architecture

4.1 Chapter Overview

The methods, instruments, and research process have all been discussed in previous chapters. The architecture, design, and operation of the proposed solution are explained in detail in this chapter. This chapter's primary focus was on describing the architecture of the model. A detailed expression of the four primary CNN architectures and associated hyperparameter adjustment is provided.

4.2 Proposed Architecture

In the proposed architecture, we implemented a transfer learning approach to develop lung cancer detection. First, a pre-trained CNN model was selected, and the fully connected layers of the model were removed. A new fully connected layer with two output nodes for 'Cancer' and 'Normal' classes was added and the weights in this layer were randomly initialized for each selected pre-trained model.

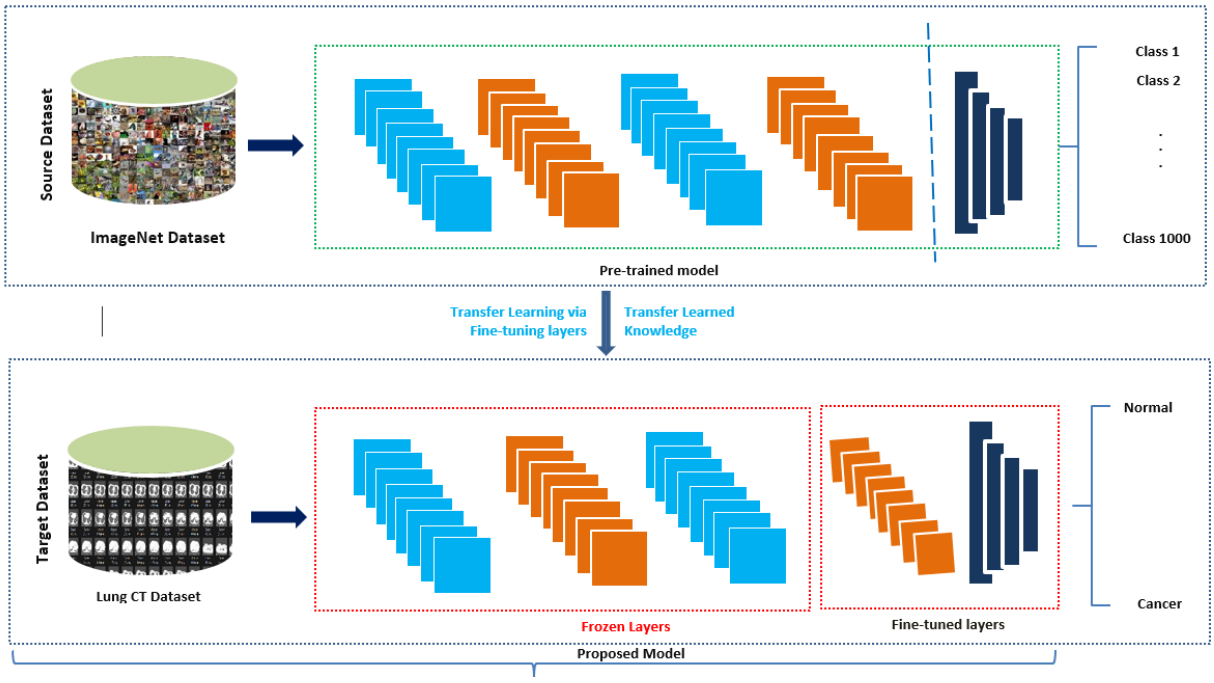


Figure 4.1 Proposed architecture for lung cancer detection

4.3 Lung Cancer Detection using DenseNet169 Fine-Tuning

In our study, we employed a transfer learning approach utilizing the pre-trained DenseNet169 model. Strong convolutional neural network architecture DenseNet169 has proven to perform magnificently in a variety of disease detection applications. We may take advantage of this pre-trained model's strong feature extraction capabilities and adapt it to the unique features of the lung cancer dataset by fine-tuning it.

The DenseNet169 model consists of dense blocks, where every layer is linked to every layer that came before it. The network's performance is enhanced and the complexity of the model is reduced due to the effective information flow and feature reuse made possible by this dense connectivity structure. 'Bottleneck' layers are another technique used by DenseNet169 to lower the processing demands and feature map count without negatively impacting the accuracy of the model. We initially frozen the weights of the pre-trained layers in order to fine-tune the DenseNet169 model for lung cancer detection. This maintains the low-level features that the model learnt up from the extensive ImageNet dataset, like edge detection and pattern recognition. The frozen DenseNet169 base model was then modified with the addition of custom layers to make it suitable for the lung cancer dataset.

By calculating the average value for every feature throughout the image, the first custom layer, called GlobalAveragePooling2D, greatly decreases the dimensionality of the feature maps. The succeeding layers become more effective and information-focused as a result of this dimensionality reduction. Subsequently, Neurons and a ReLU activation function are added to a fully linked layer. In order to provide a higher-level representation that is more appropriate for differentiating between lung images that are malignant and those that are not, this layer further processes the data. By introducing non-linearity, the Rectified Linear Unit (ReLU) activation function enables the model to learn complex correlations between the extracted features. The model utilizes a single neuron with a sigmoid activation function in the output layer. Since this is a binary classification problem (Normal vs. Cancer), the output represents the probability of the lung image belonging to the cancer class, with a value ranging from 0 (not cancer) to 1 (highly likely cancer). In this model implementation we customized hyperparameters like learning-rate, epoch, batch and like following an experiment to get better result.

Table 4 - Summary parameters of DenseNet169 in its best performance

No	Characteristics	Value
1	Total Layers	169
2	Fine-tuned layers	100
3	Total Parameter of DenseNet169	12,856,258
4	Total Trainable Parameter	3,328,0661
5	Non-trainable Parameters	9,528,192

4.4 Lung Cancer Detection using InceptionV3 Fine-Tuning

Selecting the appropriate deep learning model can be crucial for medical image processing. We therefore made the decision to investigate the possibilities of the InceptionV3 architecture for our lung cancer detection research in addition to DenseNet169. This model offers an alternative perspective to the prior DenseNet169 method, one that might lead to fresh discoveries and increase the accuracy of our diagnoses.

The unique 'Inception' modules of InceptionV3, which enable the network to handle data at several scales at once, are essential to the system's success. Through the use of many convolutions with varying filter sizes, the model is able to pick up on small details and patterns that it might have missed otherwise. Because of the complex and varied features of lung cancer images, where minor texture alterations and spatial correlations are critical in differentiating between normal and cancerous tissue, this multi-scale method is especially well-suited.

With 48 layers of expertly constructed convolutions, pooling operations, and completely connected units, InceptionV3 also has a good depth. With this depth, the model can progressively acquire increasingly abstract, high-level concepts that are necessary for successful lung cancer detection, starting with low-level features like edges and textures. This allows the model to gradually build up a complex, hierarchical representation of the input data. When it comes to fine-tuning InceptionV3 for our specific task, we followed a similar approach to the DenseNet169 model. We began by freezing the weights of the pre-trained layers, ensuring that the model's robust, general-purpose feature extraction capabilities remained intact. Then, we added a few custom layers on top, tailored to the needs of our lung cancer dataset.

First, we incorporated a GlobalAveragePooling2D layer, which allowed us to significantly reduce the dimensionality of the feature maps while preserving the most relevant information. This not only improved the efficiency of the subsequent layers but also helped the model focus on the most salient aspects of the lung images. Next, we added a fully connected layer with neurons and a ReLU activation function. This layer played a crucial role in transforming the features into a higher-level representation that was more suitable for the binary classification task at hand - distinguishing between normal and cancer lung nodules.

Table 5 - Summary of parameters of InceptionV3 in its best performance

No	Characteristics	Value
1	Total Layers	159
2	Fine-tuned layers	100
3	Total Parameter of DenseNet169	22,065,314
4	Total Trainable Parameter	13,886,594
5	Non-trainable Parameters	8,178,720

4.5 Lung Cancer Detection using Xception Fine-Tuning

Xception is built on the core principles of the Inception module, unlike the previous techniques we looked at. However, Xception adds a twist: depth wise separable convolutions, a novel idea, are used in place of conventional convolution procedures. Through the use of this technique, a standard convolution is split into two distinct steps: the first is a depth-wise convolution, which operates independently on each input channel; the output of this convolution is then mixed with a point-wise convolution. This creative design choice offers several important advantages. Xception learns a more suitable representation of the input data by separating the spatial and channel-wise properties, which significantly reduces the number of parameters and compute requirements. This not only makes the model smaller and easier to use, but it also reduces the possibility of overfitting, which is a big issue when working with limited medical datasets.

The real brilliance of Xception lies in its richness and the complex connections among its various layers. With 71 layers, the model can build a hierarchical understanding of the input lung images gradually, starting with low-level features like edges and textures and progressing to more complex, high-level representations that are necessary for accurate cancer detection.

Upon optimizing Xception for our particular lung cancer detection assignment, we employed a methodology similar to that of the prior models. The pre-trained layers' weights were first frozen in order to maintain the model's capacity to extract reliable, broadly applicable features from the input data. Then, we layered on top a few tailored layers that were made specifically for our lung cancer dataset. We were able to significantly reduce the dimensionality of the feature maps while keeping the most essential information by using a GlobalAveragePooling2D operation as the first of these custom layers. This enhanced the effectiveness of the layers that came after it and supported the model in concentrating on the most notable features of the lung images. We then added a fully linked layer with neurons and an activation function for ReLU. This layer was essential in converting the features into a higher-level representation that was more appropriate lung cancer detection.

We included a Dropout layer, which randomly eliminated certain neurons during training, to reduce overfitting and improve the model's capacity for generalization. As a result, the model had to do so learn more reliable and general features, which improved its performance on untested data. The likelihood that the input lung image belongs to the cancer class was the output of a single neuron with a sigmoid activation function, which was the model's last component. This made it possible for us to definitively determine, at an important point in the diagnostic procedure, whether the lung image was normal or cancerous. In addition to its depth and strong feature extraction capabilities, the model's capacity to learn spatial and channel-wise features quickly makes it a promising choice for our lung cancer detection assignment.

Table 6 - Summary of parameters of Xception in its best performance

No	Characteristics	Value
1	Total Layers	71
2	Fine-tuned layers	20
3	Total Parameter of DenseNet169	21,124,1010
4	Total Trainable Parameter	7,588,906
5	Non-trainable Parameters	13,535,104

4.6 Lung Cancer Detection using ResNet50 Fine-Tuning

Over time, some deep learning architectures have shown to be dependable and have become truly essential instruments in the field of medical image processing. One such model is Resnet50, a deep convolutional neural network that has demonstrated time and time again its ability to perform complex classification tasks, including the essential task of lung cancer detection. What distinguished Resnet50 from other deep neural network architectures is its promising 'residual learning' approach, which addresses the vanishing gradient problem. With shortcut connections that bypass some layers and preserve a constant information flow throughout the network, the model is able to acquire rich, hierarchical representations of the input data without falling to the negative consequences of gradient diminishment. This decision and breadth of knowledge are critical for accurately detect lung cancer from a given Lung CT images.

As we began to optimize Resnet50 for our lung cancer detection task, we were immediately pleased by the model's ability to adapt and thrive in the various challenges posed by medical imaging. By carefully adding a few additional layers customized to our dataset's training and freezing the weights of the pre-trained layers, we were able to use Resnet50's robust feature extraction capabilities while fine-tuning it to the specific requirements of our task. The final outcome was a model that demonstrated excellent classification ability along with a notable level of generalization, a crucial attribute in the high-stakes domain of medical diagnosis.

Table 7 - Summary of parameters of ResNet50 in its best performance

No	Characteristics	Value
1	Total Layers	50
2	Fine-tuned layers	7
3	Total Parameter of DenseNet169	23,850,242
4	Total Trainable Parameter	3,678,082
5	Non-trainable Parameters	20,172,160

Chapter Five

5. Implementation of the Proposed Solution

5.1 Chapter Overview

The proposed model's implementation is discussed in this chapter, including setting up the environment, preprocessing the data, augmenting the data, and training the model. Moreover, it includes explanations of code sample snippets to simplify a code understanding for a reader.

5.2 Working Environments

The hardware tool specifications and environmental setup needed to implement the model are described in this section. An essential component of the suggested solution is the working environments. This covers the software and hardware that were employed in the research. For effective data processing, a computer with a high-performance CPU and GPU is part of the hardware. Efficient creation, training, and assessment of the suggested method for CT scan image-based lung cancer detection are made possible by the following environments.

- **Laptop:**
  - Operating System: Windows 11Pro N
  - Processor: Intel® Core™ i5-8350U CPU @ 1.70GHz 1.90 GHz
  - Installed memory (RAM): 16.00 GB
  - Graphics: Intel® UHD Graphics 620
  - System Type: 64-bit operating system, x64-based processor
- **Desktop:**
  - Operating System: Windows 11 Enterprise
  - Processor: Intel(R) Core™ i7-8700 CPU @ 3.20GHz 3.19 GHz.
  - Installed memory (RAM): 8.00 GB
  - Graphics: Intel® UHD Graphics 630
  - System Type: 64-bit operating system, x64-based processor
- **Graphics Processing Unit (GPU):**
  - Graphics: NVIDIA GeForce RTX 2070 with 8 GB RAM

5.3 Environmental Setup

To set up the environment for our system, we utilized several tools. Anaconda, an open-source package manager, was employed to simplify the setup of Python and its scientific libraries. Additionally, Jupyter Notebook, a web-based application, was used to facilitate coding and real-time visualization. Python, a user-friendly and high-level programming language, served as the primary language for our system's implementation.

- **Anaconda**

One of the work environments used in CT scan image-based deep learning-based lung cancer detection is Anaconda. It's an open-source utility that makes managing packages easier and is frequently used to install Python and its different modules. The solution we propose is implemented using Anaconda version 1.9.7.

- **Jupyter Notebook**

Another workspace used in deep learning-based CT scan image-based lung cancer detection is Jupyter Notebook. It's an open-source application that combines computer output, explanatory text, software code, and multimedia assets onto a single page. The project uses Jupyter Notebook for experimentation and code development. The researcher may effortlessly compose, test, and edit code with ease, and record the procedure by incorporating text and multimedia materials. Because of this, Jupyter Notebook is the perfect tool for creating and evaluating our proposed solution.

- **TensorFlow**

TensorFlow is an open-source Python library for numerical computation by Google. It is used for developing and training Deep Learning models. TensorFlow is flexible and powerful. It can also run from a highly distributed system to a low distributed system. It is used as a back end for the Keras interface in this study to develop the model, specifically for building the proposed functional model and implementing GradCAM based interest region visualization.

- **Keras**

Keras is a high-level library written in python programing language and can run on Theano, TensorFlow, Microsoft Cognitive (CNTK)Toolkit, and PlaidML back end. It is used to build Machine learning models. Keras is used as an interface with TensorFlow back ends for developing the proposed model. Functional API offered by Keras was used to create flexible models since it supports multiple outputs and sheared layers.

5.4 Implementations

In this section we show implementation of data augmentation, data processing model developing and training, and Grad-CAM based visualization. They are shown below with description.

5.4.1 Dataset Loading

First, the dataset was prepared and stored in the code2tash folder, which contained the Normal and Cancer subfolders. Dynamic dataset splitting was used during training by employing an on-the-fly approach to data partitioning, allowing the algorithm to adaptively select samples from these subfolders. This approach improves model performance and generalization by increasing its effectiveness of data consumption and ensuring that the training and validation sets are representative of the actual data distribution. It is shown Figure 5.6 below

```
data_dir = "D:/GradCAM/code2tash"
try:
    train_df, valid_df, test_df = split_data(data_dir)
    train_gen, valid_gen, test_gen = create_gens(train_df, valid_df, test_df, batch_size)
except:
    print("invalid input!")
```

Figure 5.1 Dataset Loading implementation

5.4.2 Data Augmentation Implementation

To improve model generalizability, we employed image augmentation techniques including rotations, flips, brightness adjustments, zooming and others as shown image 5.2 below. These techniques do not simply create 9x more images per class, but rather introduce a significant variety in the training data. This variety allows the model to learn features that are robust to small variations in image appearance, which can be crucial for real-world applications. In deep learning, data augmentation shines as a powerful tool, particularly for tasks with limited datasets. To compensate for the original dataset's size, we implemented various augmentation techniques. These techniques include randomly flipping images, adjusting brightness levels, translating images horizontally or vertically, zooming in or out on specific regions, and rotating images at various angles. By artificially expanding the training data with these variations, we enhance the model's ability to generalize and improve its performance on unseen lung cancer detection scenarios. The following implementation shows all data augmentation transformations we used in our study as shown on Figure 5.2 below.

```
tr_gen = ImageDataGenerator(rotation_range=45,
                            width_shift_range=0.1,
                            height_shift_range=0.1,
                            shear_range=30.0,
                            brightness_range=[0.2,0.9],
                            zoom_range=[0.5,1.0],
                            horizontal_flip=True,
                            vertical_flip=true,
                            channel_shift_range=80.0,
                            fill_mode='reflect',
                           )
```

Figure 5.2 Data Augmentation implementation

- **Shifting**

From selected augmentation techniques one of them is shifting image pixels in a particular direction. There are two types of shifting augmentation horizontal and vertical shifting and both of them are implemented. The sample image implemented shifted horizontally and vertically is shown in the implementation result in chapter six on figure 6.1. The implementation returns an image pixel shifted in the x and y direction and clipped while specified new pixels in some regions.

- **Rotation**

A random rotation of the image in a certain direction in this case 45 degrees in a clockwise direction. It gives the site of an image from a different angle of a site without changing the image information. An empty area of images due to rotation is filled by the reflected pixel values.

- **Flipping**

Images are flipped in both horizontal and vertical directions by reversing rows vs columns of image pixels. This implementation results in a mirror reversal of the original image on the x or y axis based on the given parameter.

- **Zooming**

Image is randomly zoomed in a range of 0.5 to 1.0 by interpolating pixel values of images. Therefore, images show a variation in both width and height dimensions and different aspect ratios of the lung CT image.

5.4.3 Proposed Model Implementation

By adjusting a pre-trained models' layer, in our implementation, based on our experimental configuration, we were able to effectively solve the binary classification problem by combining a sigmoid activation function with binary cross-entropy. For the purpose of evaluating their effect on model performance, we conducted experiments with two optimizers: Adam and Adamax. A robust and efficient training procedure that was tailored to our particular dataset and targets was ensured by testing different learning rates and implementing dropout regularization (i.e. L1) to reduce overfitting as shown on Figure 5.3 below.

```
BATCH_SIZE = 16
EPOCHS = 30
def incep(fine_tune=100):
    conv_base = InceptionV3(include_top=False, weights='imagenet', input_shape=(IMAGE_SIZE, IMAGE_SIZE, 3))
    if fine_tune > 0:
        for layer in conv_base.layers[:-fine_tune]:
            layer.trainable = False
    else:
        for layer in conv_base.layers:
            layer.trainable = False

    x = tf.keras.layers.UpSampling2D(size=(4, 4))(conv_base.output)
    x = tf.keras.layers.GlobalAveragePooling2D()(x)
    x = tf.keras.layers.BatchNormalization()(x)
    x = tf.keras.layers.Dropout(0.5)(x)
    x = tf.keras.layers.Dense(256, activation='relu')(x)
    x = tf.keras.layers.BatchNormalization()(x)
    x = tf.keras.layers.Dropout(0.5)(x)
    out_layer = tf.keras.layers.Dense(1, activation='sigmoid', name='actiation')(x)

    model = Model(inputs=conv_base.input, outputs=out_layer)
    optimizer = tf.keras.optimizers.Adam(learning_rate=0.01, epsilon=1e-2)
    model.compile(loss='binary_crossentropy', optimizer=optimizer, metrics=["accuracy"])
    model.summary()
    return model
```

Figure 5.3 Model implementation

5.4.4 Early Stop Implementation

For the training of our detection model, we employed early stopping with a minimum delta of 0.001 and patience of 5 epochs. With this setup, we were able to halt the training process after up to five epochs without seeing any significant reduction in validation loss and improvement is training validation. We were able to maintain the best model while successfully controlling overfitting because the minimal delta made sure that only significant changes were considered as shown on Figure 5.4 below.

```
# Early Stopping Callback
   early_stopping = tf.keras.callbacks.EarlyStopping(
       monitor='val_accuracy',  # Monitor validation accuracy
       min_delta=0.001,         # Minimum improvement threshold
       patience=5,              # Number of epochs to wait for improvement
       mode='max',              # Track the 'max' of the monitored metric
       verbose=1                # Print information at each epoch
   )
```

Figure 5.4 Early stop implementation

5.4.5 GradCAM implementation

This code snippet shows a Grad-CAM configuration. Our base model which is our pre-trained model, is specified and the saved model is loaded for inference. Additionally, we adjust the size of the input image to match with the dimensions used in the training process. Additionally, as Grad-CAM visualization is layer-specific and helps highlight the significant elements pertinent to the model's predictions, we load the testing image that we wish to evaluate and determine the layer for it as shown on Figure 5.5 below.

```
import argparse
from keras.models import load_model
args_list = [
    '--base', 'dense',
    '--model_path', 'D:/GradCAM/code2stores/model/dense169.h5',
    '--input_size', '224',
    '--img', 'D:/GradCAM/mydata2/Testing/Cancer/Cancer (565).png',
    '--layer', 'conv5_block18_1_conv', # conv5_block32_2_conv
    '--out', '/content/drive/MyDrive/mydata/GradCamResult.png'
]

parser = argparse.ArgumentParser()
parser.add_argument('--base', type=str, help='Base model to use')
parser.add_argument('--model_path', type=str, help='Path to the saved model')
parser.add_argument('--input_size', type=int, help='Input size for the model')
parser.add_argument('--img', type=str, help='Path to the image to process')
parser.add_argument('--layer', type=str, help='Layer to visualize')
parser.add_argument('--out', type=str, help='Path to save the output image')

args = parser.parse_args(args_list)
print(args.base)
```

Figure 5.5 Grad-CAM configuration implementation

Since different models require different image sizes and normalization strategies, our Grad-CAM implementation tailors image preprocessing based on the particular base model. The image is loaded and resized by the process function to fit the model's specifications. Pixel values for models such as Xception and InceptionV3 are scaled to [0, 1]. We convert RGB to BGR and subtract mean pixel values for ResNet models. The pixel values in DenseNet must be scaled to [-1, 1]. In order to provide correct predictions in the Grad-CAM visualization, we finally enlarge the image dimensions in order to prepare it for model input as shown on Figure 5.6 below.

```
def process(path, base, SIZE):
    x = keras.utils.load_img(path, target_size=(SIZE, SIZE))
    x = keras.utils.img_to_array(x)
    if base in ['vgg16', 'vgg19', 'incepv3', 'xcep']:
        x = x / 255.0
    elif base in ['res50', 'res101']:
        x -= np.mean(x, axis=(0, 1, 2))
        x = x[..., ::-1]
    elif base in ['dense', 'mob', 'incepres']:
        x = (x / 127.5) - 1.0
    else:
        raise ValueError('Invalid model specified')
    x = np.expand_dims(x, axis=0)

    return x
```

Figure 5.6 Image processing and Normalization for base models

In order to obtain the predicted class index, accuracy, and label in this Grad-CAM implementation, we first predict the class of the input image x. Next, we compute the gradients of the predicted class with respect to the selected layer by accessing its output. We obtain the spatial map and gradients for the input image by using Keras functions. By averaging the gradients, the weights for the class activation map (CAM) are determined. Therefore, we calculate the CAM by taking the spatial map and weights dot product, scaling it to the original image dimensions, normalizing it to the range [0, 1] for visualization, and applying ReLU to preserve positive values. This process can be understood from the code snippet shown on Figure 5.7 below.

```
output_mod = model.output
output_all_cats = output_mod[0]
y_c = output_all_cats[pred_index]
spatial_map_layer = model.get_layer(layer_name).output

grads_l = K.gradients(y_c, spatial_map_layer)
grads = grads_l[0]
spatial_map_and_gradient_function = K.function([model.input], [spatial_map_layer, grads])
spatial_map_all_dims, grads_val_all_dims = spatial_map_and_gradient_function([x])
spatial_map_val = spatial_map_all_dims[0]
grads_val = grads_val_all_dims[0]
weights = np.mean(grads_val, axis=(0,1))

cam = np.dot(spatial_map_val, weights)
cam = np.maximum(cam, 0)
cam = cv2.resize(cam, (x_.shape[1], x_.shape[0]), cv2.INTER_AREA)
cam = cam / cam.max()
```

Figure 5.7 Calculating gradient and spatial map

As implemented on Figure 5.8 and shown below, we generate a figure with two subplots: the first subplot displays the original image in greyscale, and the second subplot uses a jet colormap with a transparency alpha of 0.5 to overlay the Grad-CAM output over the original image. The portions of the image that the model concentrated on, emphasizing those that were most relevant to its prediction, are visible to us. Sample visualization can be seen on image 6.2.

```
# Display the original image and the GradCAM output
    fig, ax = plt.subplots(1, 2, figsize=(12, 6))
    ax[0].imshow(keras.utils.load_img(path), cmap='gray')
    ax[0].set_title('Original Image')
    ax[0].axis('off')
    ax[1].imshow(keras.utils.load_img(path), cmap='gray')
    ax[1].imshow(cam, cmap='jet', alpha=0.5)
    ax[1].set_title('GradCAM Output')
    ax[1].axis('off')
    plt.show()
```

Figure 5.8 Displaying original CT image and GradCAM Visualization

In addition to visualization, detection performance can also be seen using the code snippet in our GradCAM implementation as shown on image 5.9 below.

```
pred = model.predict(x)
pred_index = np.argmax(pred)
pred_label = labels[pred_index]
pred_accuracy = pred[0, pred_index] * 100
print(f'Ground truth: {truth} - {labels[truth]}')
print(f'Predicted: {pred_index} - {pred_label} (accuracy: {pred_accuracy:.2f}%)')
```

Figure 5.9 Detection performance on test image

Chapter Six

6. Results and Discussion

6.1 Chapter Overview

The previous chapters have provided a brief overview of the study's background, the initiative problem that motivated the research proposal, relevant literature, methodology, suggested model, and its application. This chapter contains illustrations of the findings that have been documented following the implementation of various studies. The data collected from every training session using different validation methods is shown together with a comparison of the proposed model. Successful experiment that is conducted with different hyperparameters is presented utilizing a range of visual aids, such as tables, figures, and graphs.

6.2 Data Augmentation Results

Numerous augmentation strategies have been used to expand data size, as discussed in section 5.4.1. While adding more diversity to the input so that the model perceives it as new data, the chosen augmentation helps to preserve data that can make sense for the intended purpose. Different augmentation transformations have been considered in order to get better outcomes, and the impact of each technique implementation has been observed.

Since our dataset is limited and relatively small, augmentation enabled us to significantly expand the diversity of the training data. This method allowed us to modify existing images by rotating, moving, adjusting the brightness of them and others. By making these modifications, we improve the model's ability to generalize, resulting in better performance on previously unseen data. Additionally, data augmentation reduces the risk of overfitting by making the model less dependent on specific properties of the original images. So, this technique results in a more robust and accurate model that can better handle variances in real-world situations. Outcome of different transformations employed in our augmentation implementation is shown below on Figure 6.1. Description of each transformations are already explained in section 5.4.1.

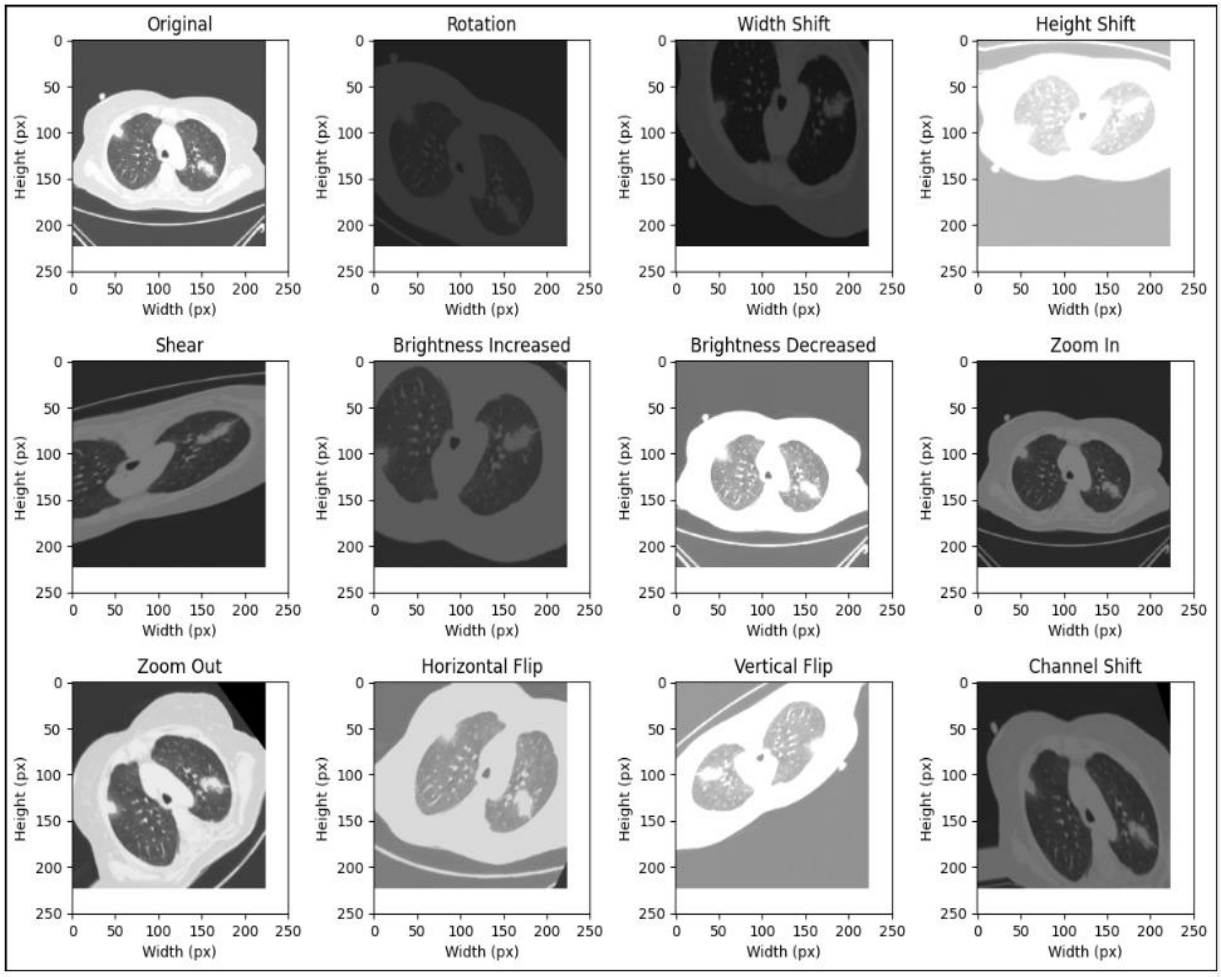


Figure 6.1 Sample data augmentation

6.3 Results of GradCAM Visualization

Understanding the focus area of an image in the model prediction's decision is extremely important in the sophisticated field of medical image analysis. In this paper the genuine potential of GradCAM (Gradient-weighted Class Activation Mapping) offers a window into the inner workings of our model for detection of lung cancer. After carefully reviewing the available visualization techniques, we determined that GradCAM would be the most effective approach to uncover the result based transparency visualization (Selvaraju et al. 2017). By generating heatmaps that highlight the regions of the input lung images that the model finds most important for a given detection, GradCAM provided us with invaluable insights. Unlike other methods, such as simple class activation mapping or layer visualization, GradCAM was able to generate these insightful heatmaps for each layer within our model (Zhou et al. 2016).

This allowed us to pinpoint the specific features and patterns that the model was leveraging to distinguish between normal and cancerous lung tissue, helping us better understand the underlying logic and refine our model further. The superior performance of GradCAM compared to these alternative techniques was a key factor in our decision to incorporate it into our lung cancer detection workflow.

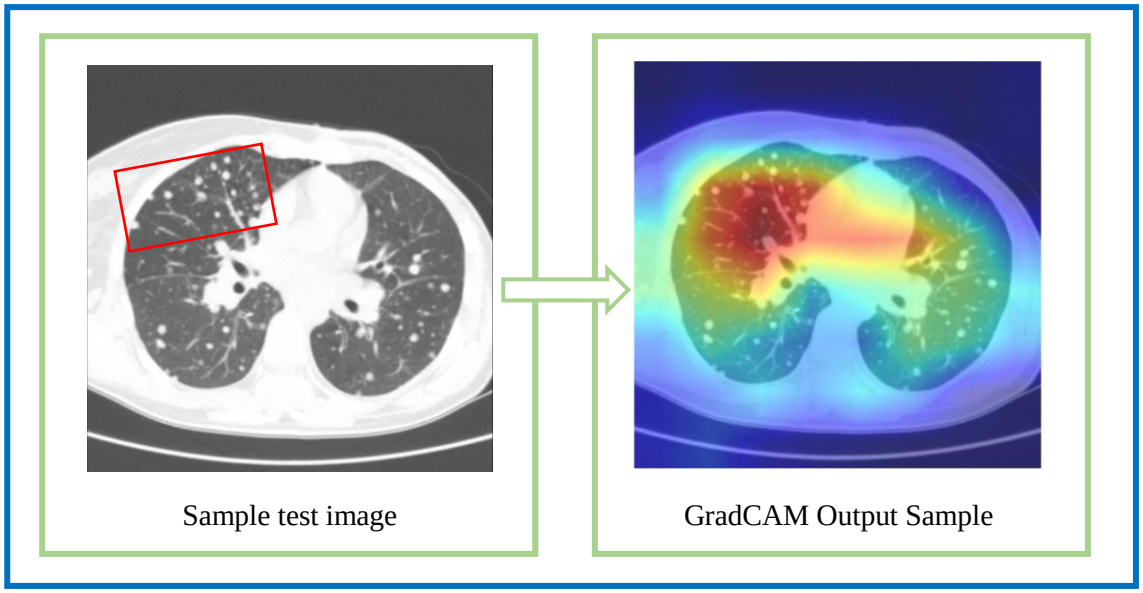


Figure 6.2 Sample GradCAM output

The following image shows original image and result of GradCAM after loading models trained with base models considered in our study - DenseNet169, InceptionV3, Xception and ResNet50.

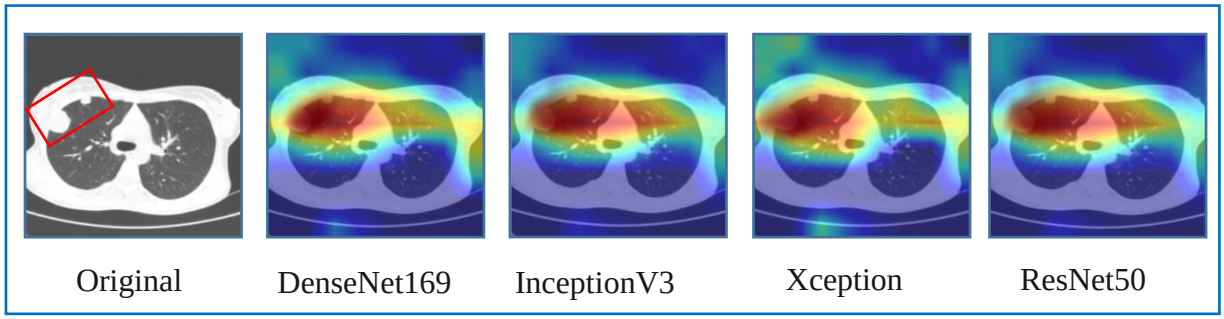


Figure 6.3 GradCAM visualization with different base models

6.4 Results of LIME

Understanding the focus area of an image in the model prediction's decision is crucial in the sophisticated field of medical image analysis. In this topic, we explore the significant potential of LIME (Local Interpretable Model-Agnostic Explanations) to shed light on the decision-making process of our model for detection of lung cancer. After reviewing various XAI techniques, we determined that LIME would be an effective method to provide local explanations for our model's predictions (Ribeiro, Singh, and Guestrin 2016). By varying the input data and tracking the subsequent adjustments to the model's predictions, LIME helps us identify the lung imaging aspects that have the most influence on the model's conclusions. This method yields understandable explanations that can be customized to each prediction, providing information about the precise regions of an image that influence the model's results.

We were able to test the model's logic for distinguishing between normal and cancerous lung tissue by using LIME, which improved our comprehension of the model's decision-making procedure. The capacity to explain model predictions helps build confidence and promote collaboration between AI systems and healthcare personnel, which makes this skill very useful in a clinical setting. Our decision to incorporate LIME into our lung cancer detection workflow was largely influenced by how well it performed at delivering usable insights into the predictions made by our model.

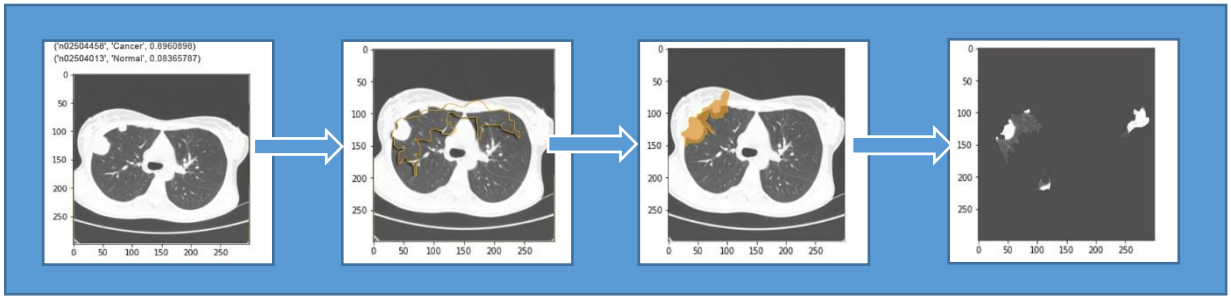


Figure 6.4 Sample LIME output

In Figure 6.4 above, the first image shows an input image and its classification accuracy within the respected base model, while the second image shows LIME based identified image Superpixel, while the third image shows shaded region of the superpixel and on the last image it cropped and identified the superpixel and shown on a gray background for better visualization and transparency.

6.5 Experiment results and Discussions

On the training, our experiment is based on the hyperparameters tuning and comparison of different base models' performance on our dataset. Accordingly, we will discuss the outcome of our training, testing and visualization with mentioned pretrained models.

6.5.1 Training with DenseNet169

To extract features from images, we leveraged a pre-trained DenseNet169 model. We fine-tuned the weights it learned from a massive dataset which is ImageNet. As utilizing all layers became too complex to our dataset, we used only the last 100 layers and frozen the rest 69 layers. Also, we added new layers on top, including ReLU for activation, average pooling for dimensionality reduction, Dropout to prevent overfitting, and finally, a Sigmoid layer for cancer prediction. These new layers were specifically trained for our lung cancer detection task.

In our experiment with DenseNet169 we tested the effect of hyperparameters in the performance of our task. We evaluated various combinations of optimizers (Adam, Adamax), dropout rates (0.5), batch sizes (16), regularization (L1 regularization) and learning rates (0.01, 0.001) within 30 epochs, all using the ReLu activation function. According to our experiment the accepted combination is when hyperparameters are set as, learning rate of 0.01 with Adam optimizers with an accuracy of 94.85%, Precision of 93.42%, Recall of 94.67% and F1-score of 94.04%. An achieved Recall result of 94.67% indicates that the model performed best in detecting lung cancer from a given CT scan image. Also, the result of F1-score, 94.04%, indicates that the model fits the balance between Normal and Cancer classes and correctly classify most of them.

To prevent the model performance from overfitting careful configuration of hyperparameters are set, drop out and regularization is implemented, trained at different epochs and early stop is included. The model training should be stop if the validation accuracy didn't improve within some number of patience. According to our training we believed that if the validation's accuracy didn't improve within five epochs, it will be halted, so we will change some hyperparameters and try a next experiment. The training graph of DenseNet169 over our dataset is visualized as the following graphs on Figure 6.5 and 6.6. As indicated on the Figure the model training is improving over an epoch. This means DenseNet was extracting meaningful information for the detection task as the number of epochs increases. Also, as the number of epochs increases the loss curve decreases.

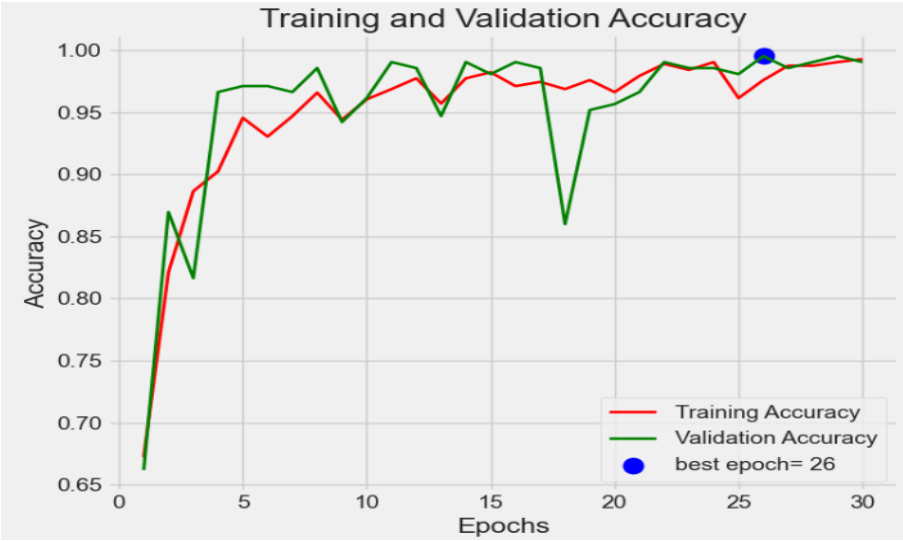


Figure 6.5 Training accuracy and Validation accuracy of DenseNet169

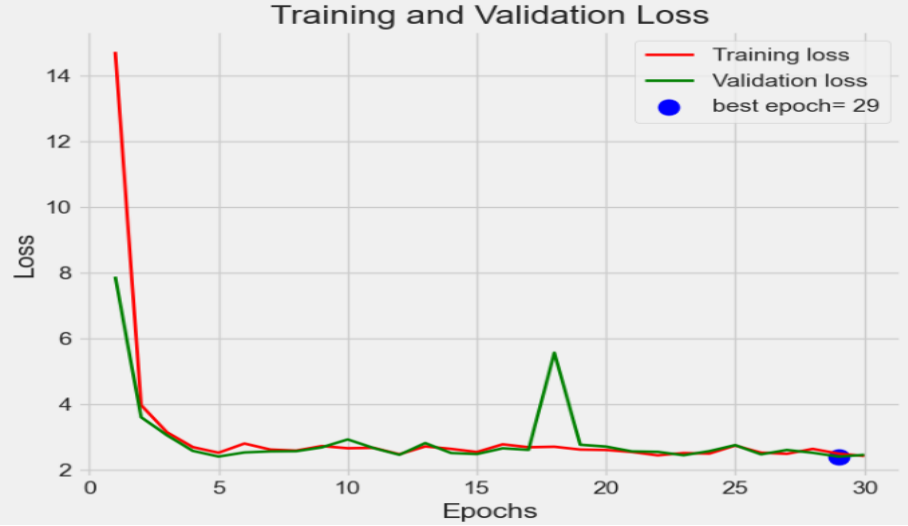
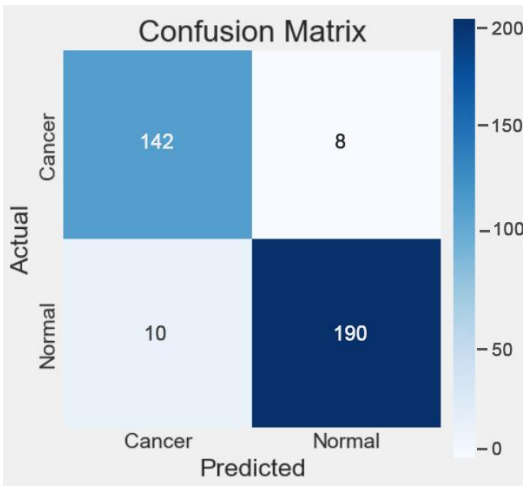
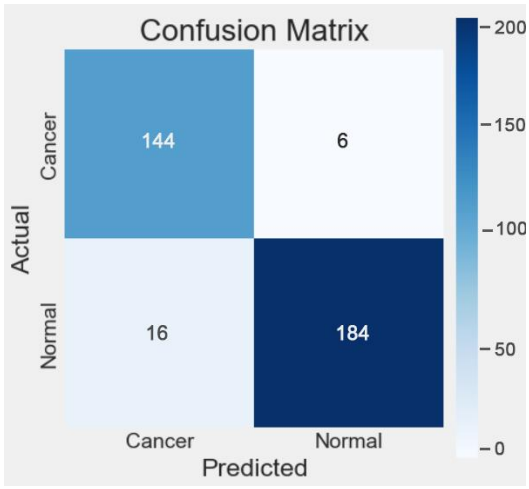


Figure 6.6 Training loss and Validation loss of DenseNet169

As DenseNet169 is a type of deep neural network architecture that is known for its efficient and robust design, it performed well on our model with an accuracy of 94.85%. However, DenseNet169 is primarily designed for colored images, it has a capability of extraction representative information from grayscale images and adapted to a new dataset with appropriate modification of an implementation. This graph, which was trained with DenseNet169, can be boldly explained by stating that epoch 26 is the best epoch for validation accuracy and epoch 29 is the best epoch for validation loss. Consequently, epoch 30 has been considered to be the best epoch since they fall within the same range as our early stop's patience.



(a)



(b)

Figure 6.7 Confusion Matrix of Proposed DenseNet169 (a) with Adam, (b) with Adamax

A comprehensive understanding on the performance metrics of the Adam and Adamax optimizers in lung cancer detection can be obtained from the confusion matrix data. The model found 190 true negatives (TN), 8 false negatives (FN), 10 false positives (FP), and 142 true positives (TP) using the Adam optimizer. This yields 93.42% precision, a 94.67% sensitivity (or recall), and a about 94.04% F1 score. The Adamax optimizers, on the other hand, produced 184 true negatives, 6 false negatives, 16 false positives, and 144 real positives. As a result, it has a 90.0% precision, a 96.0% recall, and a 92.90% F1 score.

All things considered, the comparison reveals clear advantages for each of the two optimizers. The Adam optimizers shows a strong ability to reduce false positives while keeping a high true positive rate, as evidenced by its exceptional precision and F1 score. On the other hand, the Adamax optimizers is more sensitive and can identify a higher number of lung cancer cases, although at a little reduced precision. The significance of selecting the appropriate optimizers for lung cancer detection based on particular diagnostic priorities is highlighted by these insights into precision, sensitivity, and overall accuracy.

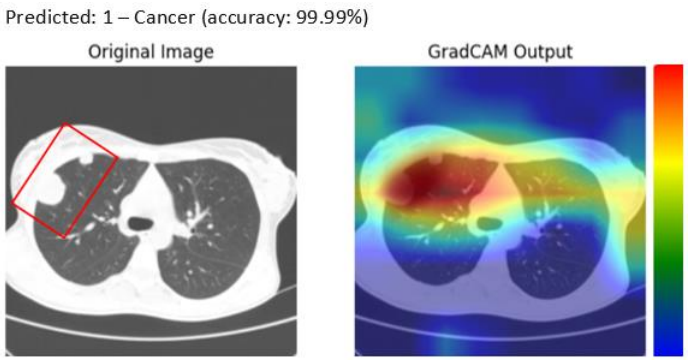


Figure 6.8 GradCAM result of CT image with a model of DenseNet169

We can deduct from Figure 6.8 that unseen image is classified as cancer with a 99.99% accuracy rate after training with Dense169. The input image is displayed on the left-hand lung CT scan, and the expertly labelled red rectangle indicates the location of the cancer nodule. The nodule location visualization based on GradCAM is thus displayed on the right-hand side, where the red color is highly heated.

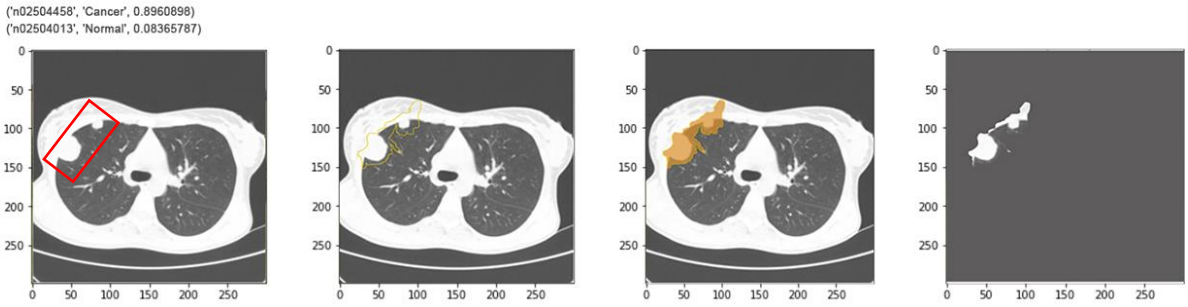


Figure 6.9 LIME result of CT image with a model of DenseNet169

As we can see from Figure 6.9, LIME performed well in explaining the most important parts of the CT lung input image that the model used to detect the input image as cancerous. As of the DenseNet169 model, an input image is identified as cancerous with an accuracy of 89.6%. LIME extracts key features (Pixels of the input image) by cropping, adding stock, and applying transparent orange color after an expert label and annotates an input image (i.e., annotation location shown using red color rectangle on input image). LIME extracts the exact pixel of an image that demonstrates the maximum degree of Explainability, in contrast to GradCAM, which applies a gradient-based heatmap to an image.

6.5.2 Training with InceptionV3

After the result of training with DenseNet169 recorded, we tried to train with InceptionV3 and compare the result. InceptionV3 has almost double amount of parameter compared to DenseNet169. So, we managed to fine-tune only desired layers of InceptionV3 for our task. In our training we experimented with different parameters and convinced with the last 100 layers from a total of InceptionV3's 159 layers, with contained almost thirteen million parameters, so that our model will not experience overfitting. Like DenseNet169, we configured different hyperparameters with suitable value to prevent overfitting.

Since a number of epochs bring the model to different performance we trained InceptionV3 at different epochs and convinced with 30 epochs like DenseNet169. At epoch 30, it achieved a better result with an accuracy of 93.14%, Precision of 90.0%, Recall of 93.4%, and F1-score of 91.69%. To achieve this result hyperparameters are set in similar way with DenseNet169 which is learning rate of 0.01, L1 regularization and drop out of 0.5, but it performed better with Adamax optimizer.

The training graph of InceptionV3 over our dataset is visualized as the following graphs on Figure 6.10 and 6.11. As indicated on the Figure the model training is improving over an epoch. This means InceptionV3 was extracting meaningful information for the detection task as the number of epochs increases. Also, as the number of epochs increases the loss curve decreases.

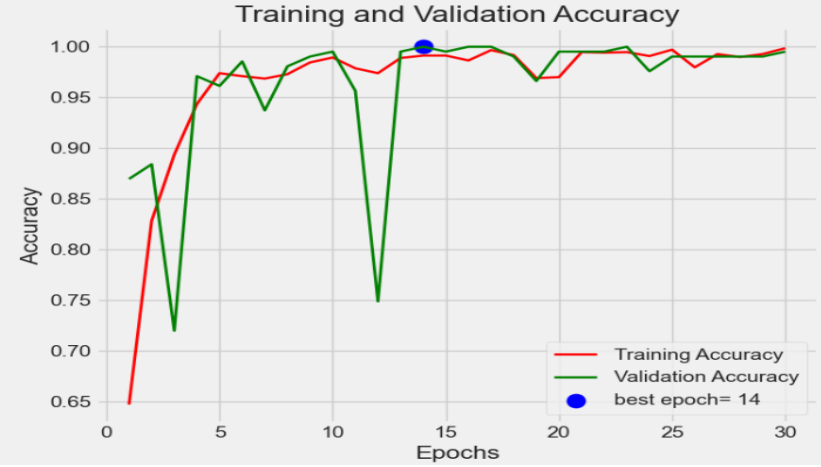


Figure 6.10 Training accuracy and Validation accuracy of InceptionV3

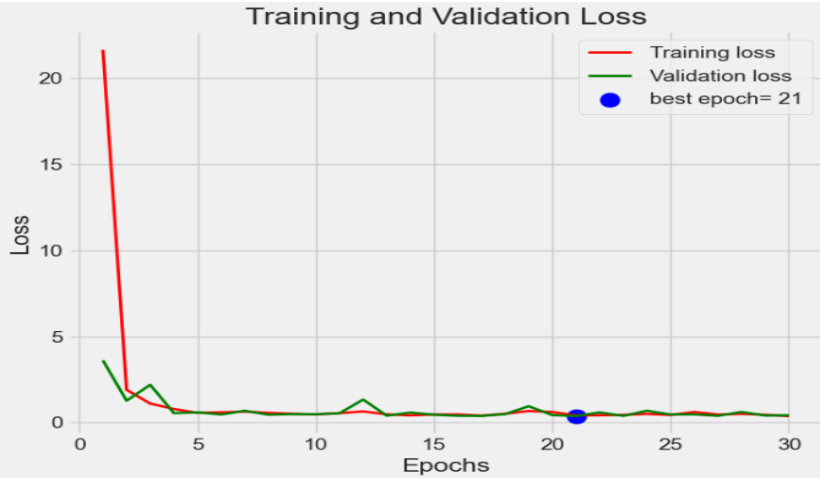


Figure 6.11 Training loss and Validation loss of InceptionV3

InceptionV3 is also considered as one of the outperforming models in medical image analysis and feature extraction in grayscale images. As per our task, it brought promising result with a Recall of 93.4% which shows its understanding of cancer features and Precision of 90.0%, so that non-cancer images are also highly recognized as a normal.

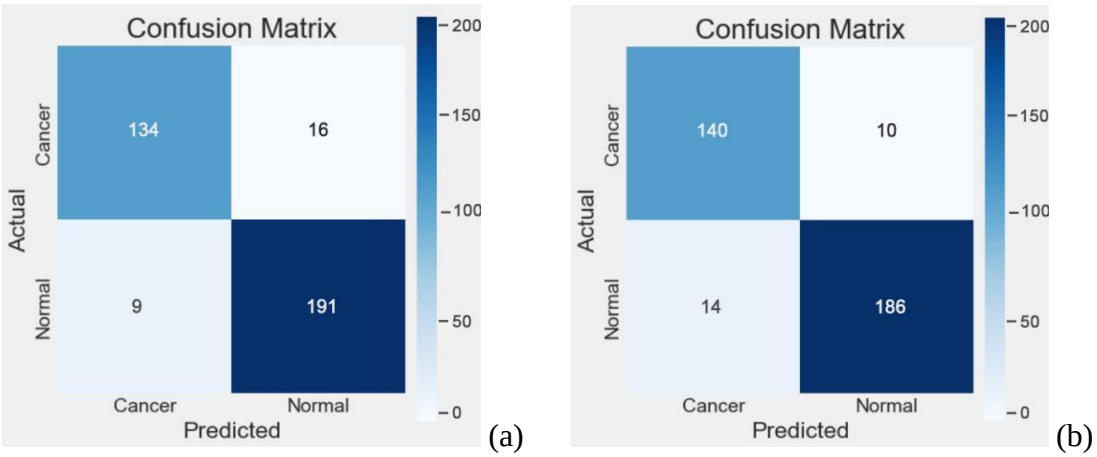


Figure 6.12 Confusion Matrix of Proposed InceptionV3 (a) with Adam, (b) with Adamax

The InceptionV3 model's confusion matrix findings show that both optimizers function well, though they each have their own advantages. The Adam optimizers produced results with a precision of 93.7%, recall of 89.34%, and accuracy of 92.85%. By comparison, the Adamax optimizers achieved approximately 93.14% accuracy, 90.0% precision, and 93.4% recall. Overall, the Adamax optimizers performs better in terms of recall and overall accuracy, while the Adam optimizers shines in terms of precision. This analysis emphasizes how crucial it is to choose the right optimizers for lung cancer detection based on certain diagnostic objectives.

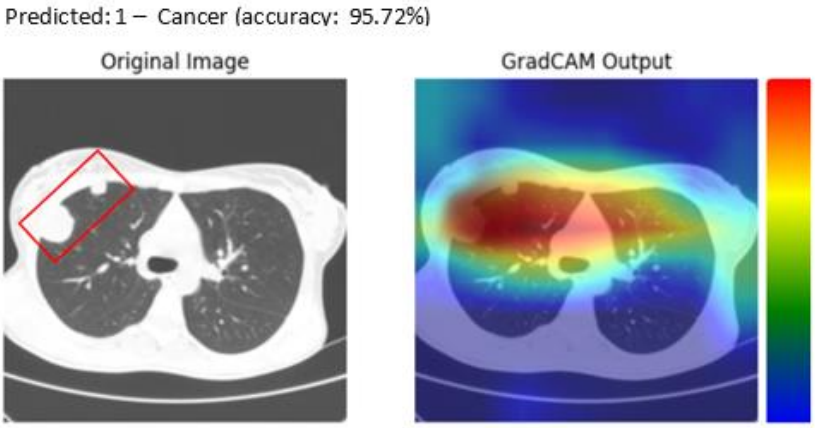


Figure 6.13 GradCAM result of CT image with a model of InceptionV3

We can understand from Figure 6.13 that a new image is classified as cancer with a 95.72% accuracy rate after training with InceptionV3. The input image is displayed on the left-hand lung CT scan, and the expertly labelled red rectangle indicates the location of the cancer nodule. The nodule location visualization based on GradCAM is thus displayed on the right-hand side, where the red color is highly heated.

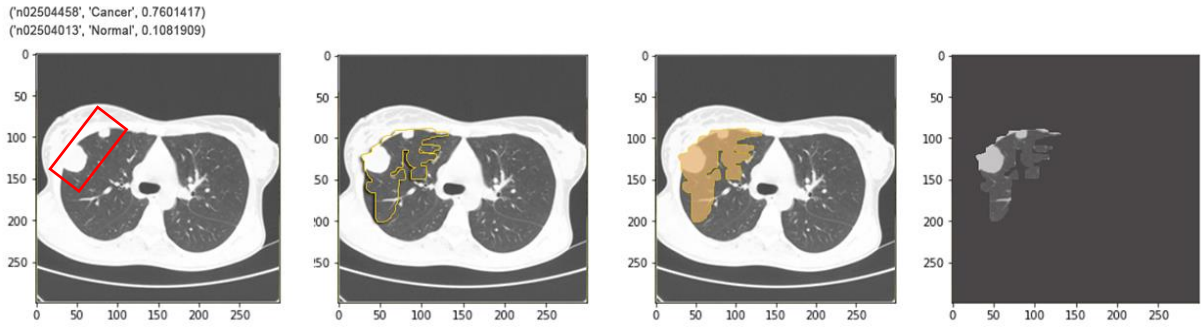


Figure 6.14 LIME result of CT image with a model of InceptionV3

As we see from figure 6.14 InceptionV3's LIME explanation achieves an accuracy of 76% in detecting a new image as cancerous, compared to DenseNet169's 89.6% accuracy. In this model, the extracted feature of LIME with this model covers a wider range of locations than DenseNet169 and includes some neighboring regions of the scan as an expert annotation indicates.

6.5.3 Training Result with Xception

Subsequent to recording the training results with DenseNet169 and InceptionV3, we sought to explore Xception and draw a comparison. Interestingly, Xception has a comparable parameter count to InceptionV3. Hence, we selectively fine-tuned the desired layers of Xception for our specific task. Throughout our training process, we experimented with diverse parameters and ultimately settled on the final 20 layers out of 71 total layers, which encompassed approximately seven million parameters - a configuration that effectively mitigated the risk of overfitting. Akin to our approach with InceptionV3, we carefully calibrated various hyperparameters to strike a balance between underfitting and overfitting. Similar to DenseNEt169 and InceptionV3, we trained Xception at 30 epochs and were rewarded with a commendable performance - an accuracy of 93.71%, Precision of 91.55%, Recall of 94.0%, and an F1-score of 92.76%. To achieve these results, we aligned the hyperparameters with those used for InceptionV3, including a learning rate of 0.01 with the Adam optimizer, L1 regularization, and a dropout rate of 0.45. The training progress of Xception on our dataset is visualized in Figures 6.15 and 6.16. As evident from these plots, the model's training performance steadily improved over the course of the epochs, indicating that Xception was effectively extracting meaningful information for the detection task. Correspondingly, the loss curve exhibited a downward trend as the number of epochs increased.

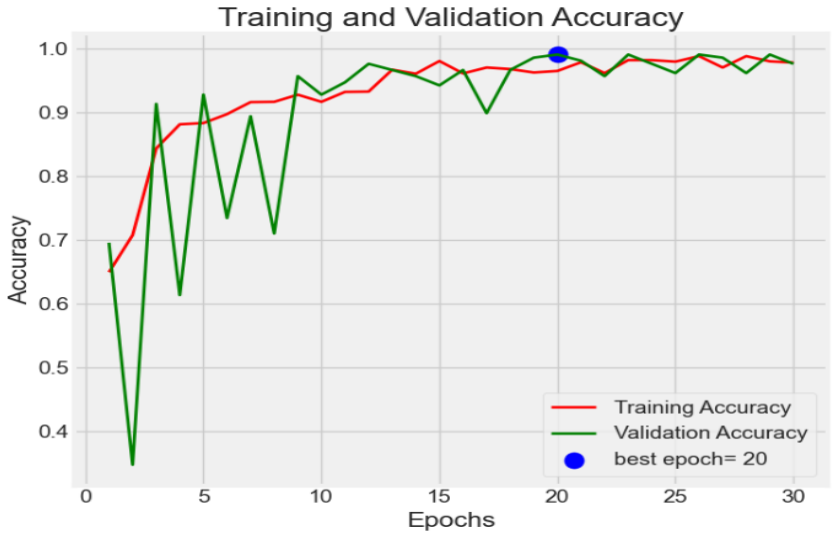


Figure 6.15 Training Accuracy and Validation Accuracy of Xception

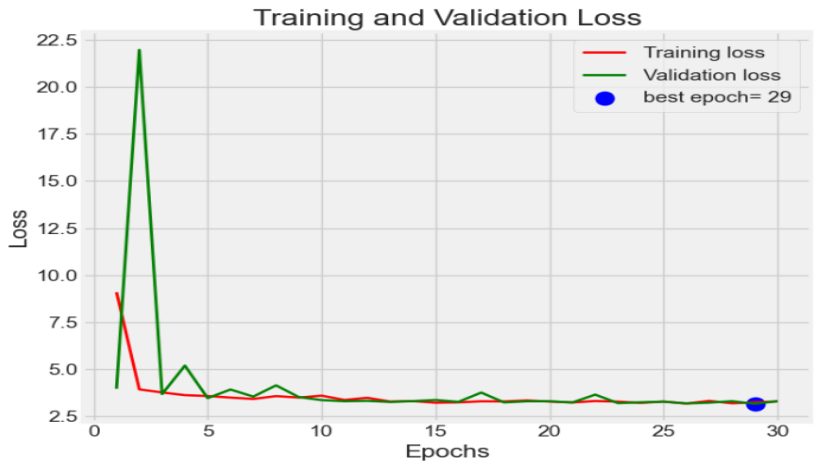
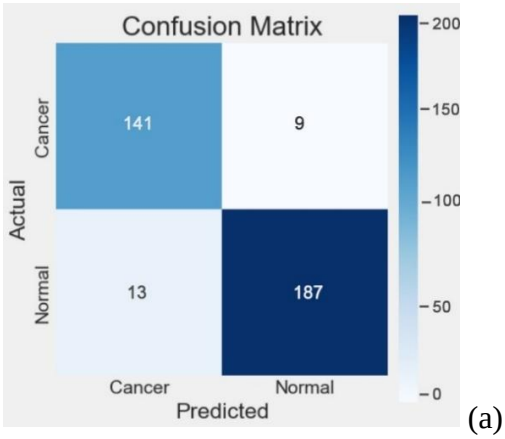
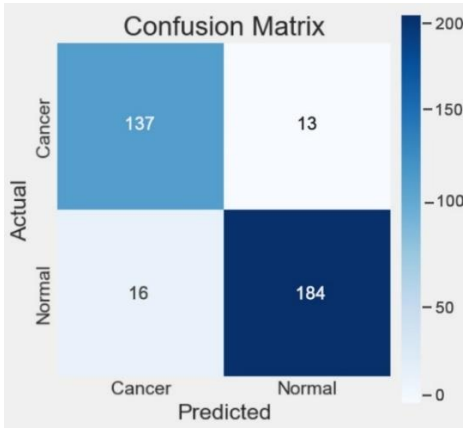


Figure 6.16 Training loss and Validation loss of Xception



(a)



(b)

Figure 6.17 Confusion Matrix of Proposed Xception (a) with Adam, (b) with Adamax

The Xception model's confusion matrix results show that the two optimizers have different performance traits. While misidentifying 9 lung cancer cases (FN) and misclassifying 13 normal cases (FP), the Adam optimizers successfully recognized 141 lung cancer patients (TP) and accurately classified 187 normal cases (TN). With a precision of 91.55% and a recall of 94.0%, this yields an approximate accuracy of 93.71%. On the other hand, the Adamax optimizers missed 13 lung cancer cases (FN) and incorrectly classified 16 normal cases (FP), correctly identifying 137 lung cancer cases (TP) and 184 normal cases (TN). This results in a precision of 89.54%, recall of 91.34%, and accuracy of almost 91.71%.

When compared to the Adamax optimizers, the Adam optimizers performs better overall in terms of accuracy, precision, and sensitivity. The Adam optimizers turns out to be the more sensible option in this case, underscoring the significance of choosing the best optimizers for lung cancer detection based on particular diagnostic aims.

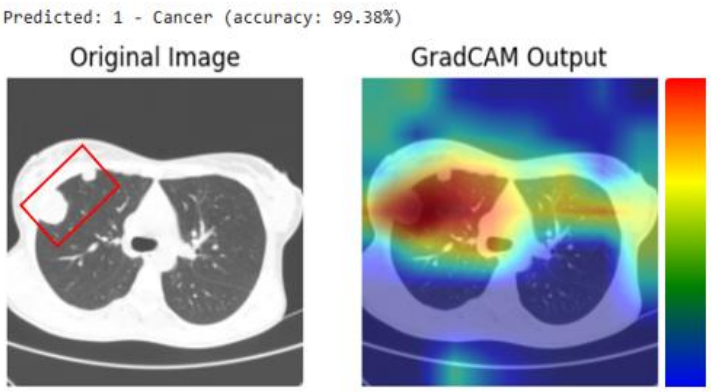


Figure 6.18 GradCAM result of CT image with a model of Xception

As shown on Figure 6.18 and similar to both previous model's GradCAM visualization and classification accuracy, a new image is classified as cancer with a 99.38% accuracy rate after training with Xception. This can be considered as a better understanding where the input image is displayed on the left-hand lung CT scan, and the expertly labelled red rectangle indicates the location of the cancer nodule. The nodule location visualization based on GradCAM is thus displayed on the right-hand side, where the red color is highly heated. As Xception's performance on the validation data shown a promising performance it also achieved high accuracy in unseen data, which shows its best performance in lung cancer detection.

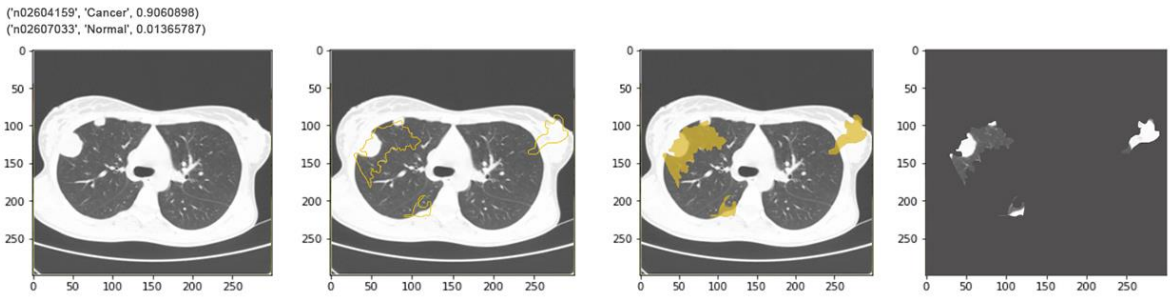


Figure 6.19 LIME result of CT image with a model of Xception

Figure 6.19 also conveys the models' detection process, which is explained by LIME. As we can see, Xception's LIME explanation achieves an accuracy of 90.6% in detecting a new image as cancerous, which is high compared to both previous models. Unlike LIME's performance in the previous models, in this model three extracted pixel locations are shown and still outperformed in showing cancerous nodules even hidden in the lung bone.

6.5.4 Training result with ResNet50

Following the promising results, we achieved with DenseNet169, InceptionV3 and Xception, our next step was to explore the potential of ResNet50 for our task. So, ResNet50 has considerably larger parameter count compared to the previous three models we investigated which is around 24 million. Nonetheless, we were determined to uncover whether this more compact architecture could deliver comparable, or even superior, performance.

Our training regimen with ResNet50 involved a meticulous process of parameter tuning and experimentation. Ultimately, we found that fine-tuning the final 7 layers of the network out of 50 total layers, which encompassed approximately 4 million parameters, struck the optimal balance between model complexity and the risk of overfitting. Consistent with our previous approaches, we meticulously adjusted the hyperparameters to ensure the model's robust generalization capabilities, steering clear of both overfitting and underfitting.

The training of ResNet50 was conducted over the course of 30 epochs, and we were delighted to witness the model's steady improvement in performance. The final results were truly impressive, with ResNet50 achieving an accuracy of 91.14%, Precision of 87.89%, Recall of 92.0%, and an F1-score of 89.89%. To attain these remarkable outcomes, we tailored the hyperparameters, settling on a learning rate of 0.01, the Adam optimizer, L1 regularization, and a dropout rate of 0.45. The training dynamics of ResNet50 are depicted in Figures 6.20 and 6.21. As the plots clearly demonstrate, the model's training performance steadily ascended with each passing epoch, showcasing ResNet50's remarkable ability to extract meaningful features from our dataset. Correspondingly, the loss curve exhibited a consistent downward trend, further validating the model's learning prowess.

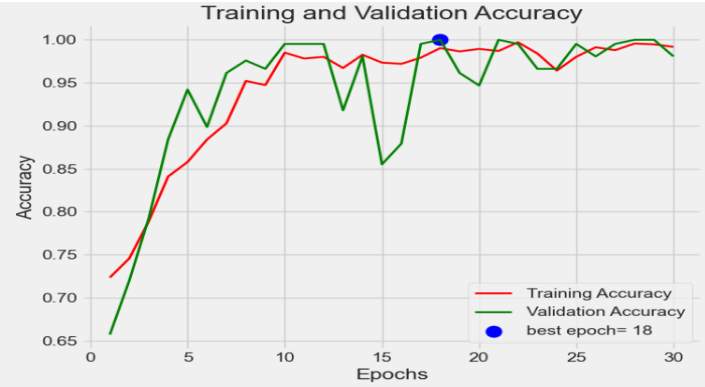


Figure 6.20 Training Accuracy and Validation Accuracy of ResNet50

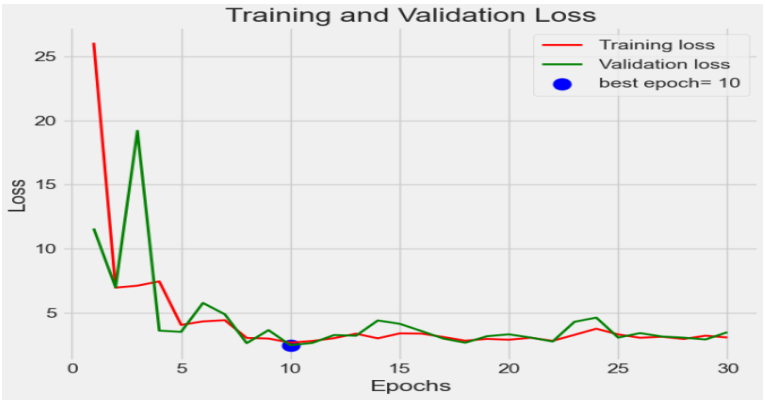


Figure 6.21 Training Loss and Validation loss of Xception

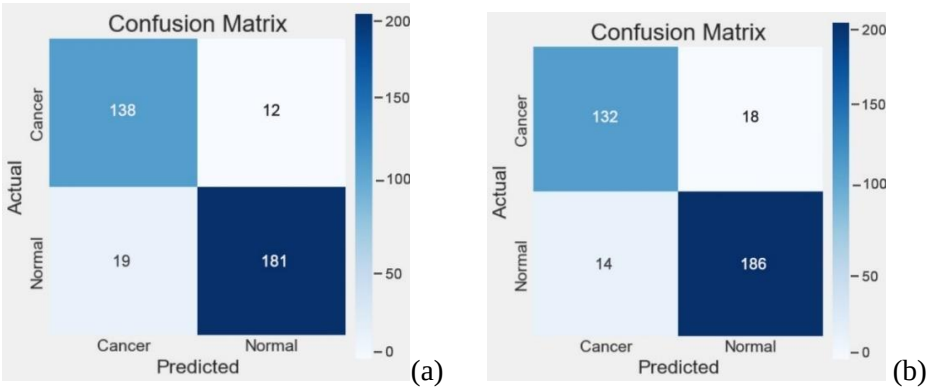


Figure 6.2215 Confusion Matrix of Proposed ResNet50 (a) with Adam, (b) with Adamax

The ResNet50 model's confusion matrix results show distinguishing variables that influence the two optimizers' performance. While the Adam optimizers misclassified 19 normal cases (FP) and missed 12 lung cancer cases (FN), it correctly identified 138 cases of lung cancer (TP) and accurately classified 181 cases of normalcy (TN). As a result, there is 91.14% accuracy, 87.89% precision, and 92.0% recall. On the other hand, the Adamax optimizers missed 18 lung cancer instances (FN) and incorrectly classified 14 normal cases (FP), while correctly identifying 132 lung cancer patients (TP) and accurately classifying 186 normal cases (TN). This results in a precision of 90.41%, recall of 88.00%, and accuracy of almost 90.85%. When compared to the Adamax optimizers, the Adam optimizers performs better overall in terms of accuracy, recall and F1 score. The Adam optimizers turns out to be the more sensible option in this instance, underscoring the significance of choosing the best optimizers for lung cancer detection based on particular diagnostic aims with ResNet50.

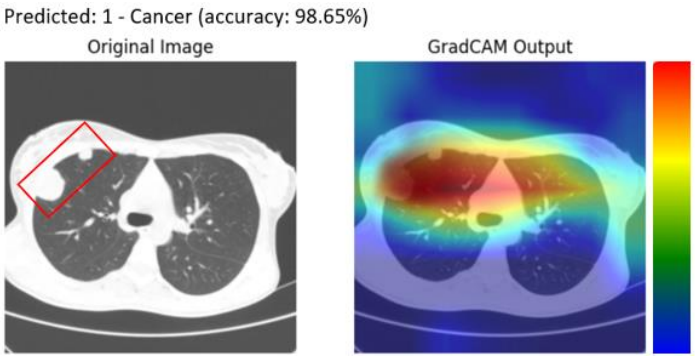


Figure 6.23 GradCAM result of CT image with a model of ResNet50

Also, in ResNet50, we applied GradCAM visualization to our new unseen data. Accordingly, GradCAMs' classification accuracy on a new image is recorded and detected as cancer with 98.65% accuracy rate after training with ResNet50. This can be considered as a better understanding where the input image is displayed on the left-hand lung CT scan, and the expertly labelled red rectangle indicates the location of the cancer nodule. The nodule location visualization based on GradCAM is thus displayed on the right-hand side, where the red color is highly heated.

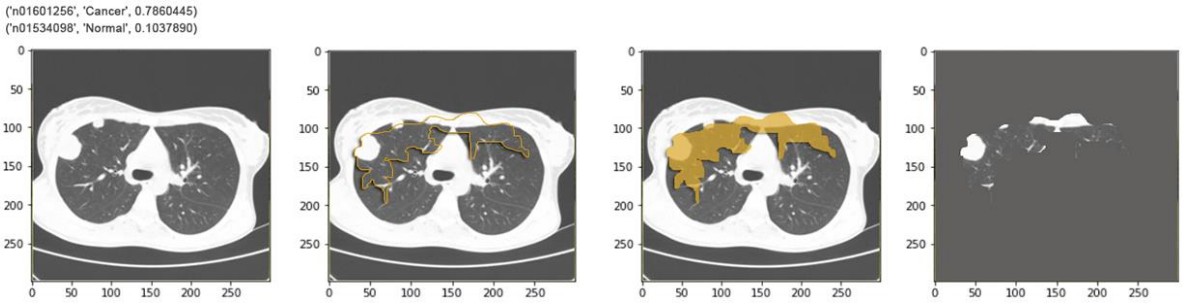


Figure 6.24 LIME result of CT image with a model of ResNet50

As we see from figure 6.24 ResNet50's LIME explanation achieves an accuracy of 78.6% in detecting a new image as cancerous. In this model, the extracted feature of LIME with this model covers a wider range of locations than other models and includes some neighboring regions of the scan as an expert annotation indicates and elongated to horizontally along the lung sides of scan.

6.6 Comparison of Proposed Architecture Results

Table 8 - Comparison of Proposed Architecture result

Model	Batch Size	Epochs	Drop out	Activ/n Function	Optimizer	Learning rate	Accuracy	Precision	Recall	F1-Score
DeneseNet169	16	30	0.5	ReLu	Adam	0.01	**94.85**	**93.42**	**94.67**	**94.04**
						0.001	90.43	91.21	90.60	90.90
					Adamax	0.01	**93.71**	**90.0**	**96.0**	**92.9**
						0.001	88.60	90.80	91.32	91.05
Inception V3	16	30	0.5	ReLu	Adam	0.01	92.85	93.7	89.34	91.47
						0.001	90.70	92.93	91.12	92.01
					Adamax	0.01	**93.14**	**90.0**	**93.4**	**91.69**
						0.001	91.19	91.0	91.34	91.17
Xception	16	30	0.45	ReLu	Adam	0.01	**93.71**	**91.55**	**94.0**	**92.76**
						0.001	89.16	92.6	90.67	91.62
					Adamax	0.01	91.71	89.54	91.34	90.43
						0.001	86.0	90.12	89.17	89.64
ResNet50	16	30	0.45	ReLu	Adam	0.01	**91.14**	**87.89**	**92.0**	**89.89**
						0.001	89.80	88.06	91.58	89.78
					Adamax	0.01	90.85	90.41	88.0	89.19
						0.001	86.25	88.20	86.21	87.19

The results of our experiments with several pre-trained models indicate outstanding performance overall, especially with DenseNet169 and Xception, which achieved accuracies of 94.85% and 93.71% using the Adam optimizer at a learning rate of 0.01. DenseNet169's performance in the lung cancer detection task was highlighted by its efficient high recall, precision, and F1-scores. Significant results were also obtained by Inception V3 and ResNet50, with accuracies of approximately 93.14% and 91.14%, respectively, particularly when the Adamax optimizer was utilized. All in all, these results indicate that although the models all performed well, DenseNet169 and Xception displayed the best overall metrics, which qualified them as good options for this detection task. We believe that the outstanding performance of DenseNet169 and Xception over other models can be credited to their advanced architectural designs, which provide resilience and improved detection accuracy by facilitating effective handling of complicated patterns in the data and promoting efficient feature reuse.

Table 9 Comparison of GradCAM and LIME's accuracy result of proposed models

No	Model	GradCAM	LIME
1	DenseNet169	99.99%	89.6%
2	InceptionV3	95.72%	76.0%
3	Xception	99.38%	90.6%
4	ResNet50	98.65%	78.6%

As we can see from Table 9, the accuracy results of each proposed model on the newly unseen data are provided. DenseNet169 identified a cancer image (as identified by professionals) in the GradCAM case with an accuracy of 99.99%, but Xception showed an accuracy of 90.6% when utilizing LIME for detection. All things considered, both GradCAM and LIME performed admirably in locating regions of interest and obtaining outstanding accuracy values in the identification of images that were cancerous.

6.7 Research questions and discussion

This study provided answers to all the research questions listed in section 1.4. So, under this section we will answer raised questions based on our experiment.

RQ-1: Which Pre-trained deep learning models are suited for lung cancer detection?

In our experiment of pre-trained deep learning models for lung cancer detection, DenseNet169 and Xception are found as the front-runners, demonstrating superior performance across multiple metrics. DenseNet169 achieved the highest accuracy at 94.85%, coupled with a precision of 93.42% and a recall of 94.67%, resulting in a commendable F1-score of 94.04. This model's architecture allows it to effectively capture intricate features in lung CT images, making it particularly adept at minimizing false negatives. Similarly, Xception showcased a robust performance, reaching an accuracy of 93.71% with a precision of 91.55% and a recall of 94.0%,

which yielded an F1-score of 92.76. Both models benefit from their ability to leverage learned features from large datasets, enabling them to generalize well to the specific characteristics of lung cancer imaging.

In contrast, InceptionV3 and ResNet50, while still effective, exhibited slightly lower performance metrics. InceptionV3 achieved an accuracy of 92.85%, with a precision of 93.7% but a lower recall of 89.34, resulting in an F1-score of 91.47. This discrepancy suggests that InceptionV3 may miss some cancerous cases, highlighting the importance of balancing precision and recall in medical diagnoses. ResNet50, on the other hand, recorded an accuracy of 91.14%, precision of 87.89%, and recall of 92.0%, leading to an F1-score of 89.89. Although it performed adequately, its overall metrics indicate that it may require further optimization to reach the effectiveness of DenseNet169 and Xception. Overall, these findings underscore the critical role of model selection and fine-tuning in enhancing diagnostic accuracy in lung cancer detection.

RQ-2: What techniques are used to reduce computational complexity of deep learning models in lung cancer detection?

We used various effective techniques of reducing the computational complexity of deep learning models in lung cancer detection. First, we fine-tuned only the top layers of pretrained models using layer freezing techniques. The top 100 layers of DenseNet169, 100 layers of InceptionV3, 20 layers of Xception, and 7 layers of ResNet50, for example, were used where DenseNet169 has 169 layers, InceptionV3 has 159 layers, Xception has 71 layers and ResNet50 has 50 layers. This method significantly decreases the number of trainable parameters, which reduces training time and computational resource requirements, while still allowing us to take advantage of the rich feature representations that deeper layers have learnt.

We faced significant problems when we tried to fully train the pre-trained models, including overfitting and underfitting, which frequently resulted in complicated training graphs and prolonged training procedures. Adding to the complexity of matters, our limited memory resources also frequently caused the training process to fail. We effectively mitigated these

problems by fine-tuning only the top layers of models such as DenseNet169, InceptionV3, Xception, and ResNet50. This strategy lowered the number of trainable parameters and accelerated the training procedure, saving us from the computational burden of fully training the model and enabling us to take advantage of the rich feature representations that deeper layers have learnt. Fine-tuning hence improved our model's performance while preserving efficiency, which finally resulted in a more reliable detection of lung cancer with less memory consumption and computational complexity.

We additionally optimized the hyperparameters to further improve the model's efficiency. We were able to achieve a compromise between model complexity and performance by carefully choosing learning rates, batch sizes, and dropout rates. In addition to accelerating convergence, this hyperparameter tuning reduced overfitting, preserving the models' resilience while they processed the complex data on lung cancer. When combined, these techniques significantly reduce training time and increase the computational viability of deep learning models for lung cancer detection.

RQ-3: How we can improve both disease detection performance and transparency?

We used a variety of cutting-edge approaches that greatly increase the efficiency of our models while maintaining interpretability in order to increase both disease detection performance and transparency. We found that applying transfer learning to pre-trained models led to significant improvements in detection performance, especially in scenarios with limited data. This method improved lung cancer detection accuracy while simultaneously speeding up training. In addition, we used data augmentation methods in order to boost the size of our training dataset, including rotation, scaling, and flipping. By lowering the chance of overfitting and enhancing the model's generalization to new data, this approach helped to enhance diagnostic results.

Grad-CAM (Gradient-weighted Class Activation Mapping) is a tool that we used to improve transparency, but it's important to acknowledge that it has key strengths of its own in terms of producing qualitative visualizations that are essential in the medical sector. Grad-CAM is an efficient tool for highlighting the precise areas of an image that have the greatest influence on the model's predictions. This makes it possible for medical practitioners to see the reasoning behind a diagnosis. This feature increases the level of trust and promotes better communication and cooperation between AI systems and medical professionals by improving the model's decision-making process. Furthermore, Grad-CAM can be very helpful in spotting minute details that conventional analysis can miss, enhancing the precision of the diagnosis. It is an essential tool for interpreting complicated models in practical applications because of its capacity to produce understandable heatmaps.

We also used LIME (Local Interpretable Model-Agnostic Explanations), which is a more quantitative approach to model interpretability and a complement to Grad-CAM. LIME uses input data perturbations and analysis of the resulting changes on predictions to produce local approximations of the model's decision boundaries. This enables us to pinpoint particular characteristics that have a big influence on model results. A clearer perspective of the decision-making process is provided by LIME, which quantifies the contribution of different aspects to the final prediction, in contrast to Grad-CAM, which might identify regions of interest. By incorporating LIME into our workflow, we improve our model's interpretability while also providing a useful information that practitioners can use to make more informed clinical decisions. Grad-CAM's visual insights and LIME's quantitative evaluations work together to provide a strong framework that enhances transparency and disease detection performance, which will ultimately result in more dependable and trustworthy AI systems in the healthcare industry.

Chapter Seven

7. Conclusion and Future Works

7.1 Conclusion

Deep learning algorithms are continually improving, revolutionizing the field of medical image processing and offering up possibilities to more reliable and precise diagnostic tools. We have noticed that convolutional neural network (CNN) architectures can perform well in the field of lung cancer detection. In this work we identified Transfer Learning's potential for lung cancer detection using the model's remarkable feature extraction and representation capabilities. By means of experimental testing and analysis, we have shown that DenseNet169 and Xception can get better results in the lung cancer detection task than other well-known CNN models, like InceptionV3, and ResNet50 with relatively small difference.

The DenseNet169 model achieved an accuracy of 94.85%, precision of 93.42%, Recall of 94.67% and F1-score of 94.04%, after being trained on a carefully annotated dataset of CT images collected from TASH. This high-performance of the model demonstrates its remarkable capacity to detect and classify the complex patterns and features linked to lung cancer, even in sophisticated medical imaging data. DenseNet169's performance in this field can be credited to its unique architectural design, which uses dense connectedness to enable effective information flow and feature reuse. This, together with the expressive ability and depth of the model, it helps to extract and capture relevant information from the input images, resulting in promising accurate predictions.

Our results highlight DenseNet169's huge ability to transform the detection of lung cancer. Our finding could improve patient outcomes and lower healthcare expenses by giving physicians a strong and dependable tool for early detection. The concepts and methods used in this research can be adapted and used to other medical imaging tasks, opening up possibilities for breakthroughs in different domains of healthcare. So, we believe that the consequences of this study go beyond lung cancer. The future is promising for even more precise and responsive imaging technologies that can greatly improve patient care, especially as deep learning and medical image processing continue to make amazing progress.

In addition to transfer learning's potential in medical image processing, visual explanation techniques are transforming the field by providing crucial insights into deep learning models' decision-making processes. These tools enhance the transparency of complex models and empower medical professionals to understand the specific regions of images that guide the model's predictions. Grad-CAM effectively highlights critical features, allowing healthcare workers to visualize areas of importance in the diagnostic process, thereby fostering trust and enhancing collaboration in clinical settings. This transparency is essential in medical applications, where the stakes are high and there is a critical need for reliable diagnostic instruments. Complementing Grad-CAM's qualitative insights, LIME (Local Interpretable Model-Agnostic Explanations) quantifies the contribution of individual features to model predictions, offering a more detailed understanding of the underlying decision-making process. By integrating these powerful visual explanation techniques with cutting-edge deep learning architectures, we can significantly enhance the performance and reliability of diagnostic tools for lung cancer detection, ultimately improving patient outcomes and increasing the acceptability of AI-assisted diagnosis in clinical practice.

7.2 Recommendations and Future Works

These days, lung cancer has become a challenge around the world and many people are dying of this disease in Ethiopia even without knowing they had a lung cancer, for it has a characteristic of being exposed at its later stage where its treatment is expensive and tough. In order to enhance the services offered and develop strategies that will prevent or restrict the spread of this illness, it is crucial to do research on the application of deep learning algorithms to clinical data of lung cancer early detection. Ethiopia falls short of technological and health care system integration, which places a great deal of responsibility on medical professionals and makes it difficult for researchers to carry out their work. To address these issues, health centers need to be improved more in order for professionals to provide patients with the right care and for researchers to carry out their work effectively.

Apart from improving the healthcare infrastructure, increasing the size of the dataset is crucial to enhancing model performance, particularly when training with full layers. So, by employing a wide variety of data gathered from several institutions, the model will be more resilient and broadly applicable, guaranteeing that it functions well in a variety of demographics.

Special Acknowledgement

We gratefully acknowledge funding support from Adama Science and Technology University under grant number:

ASTU/SM-R/1056/24

Adama, Ethiopia

References

ACS. 2024. “Key Statistics for Lung Cancer.” *American Cancer Society*: 1–12. https://www.cancer.org/cancer/types/lung-cancer/about/key-statistics.html.

AL-Huseiny, Muayed S., and Ahmed S. Sajit. 2021. “Transfer Learning with GoogLeNet for Detection of Lung Cancer.” *Indonesian Journal of Electrical Engineering and Computer Science* 22(2): 1078–86.

Algorithm, Tradaboost, Kangji Li, and Borui Wei. 2022. “A Data-Efficient Building Electricity Load Forecasting Method Based on Maximum Mean Discrepancy and Improved.”

Ali, Luqman et al. 2021. “Performance Evaluation of Deep CNN-Based Crack Detection and Localization Techniques for Concrete Structures.” *Sensors* 21(5): 1–22.

Alzubaidi, Laith et al. 2021. 8 Journal of Big Data *Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions*. Springer International Publishing. https://doi.org/10.1186/s40537-021-00444-8.

Awedew, Atalel Fentahun, Zelalem Asefa, and Woldemariam Beka Belay. 2022. “National Burden and Trend of Cancer in Ethiopia, 2010-2019: A Systemic Analysis for Global Burden of Disease Study.” *Scientific reports* 12(1): 12736. https://doi.org/10.1038/s41598-022-17128-9.

Brock, Malcolm V. et al. 2008. “DNA Methylation Markers and Early Recurrence in Stage I Lung Cancer.” *New England Journal of Medicine* 358(11): 1118–28.

Cao, Yan Ning, Xiao Chuanc Chai, Han Hui Zhang, and Qi Zheng. 2011. “Two Zinc(II) Carboxyarylphosphonates with Unusual Polynuclear Zn(II) Units: Syntheses and Crystal Structures.” *Jiegou Huaxue* 30(6): 775–84.

Chollet, François. 2017. “Xception: Deep Learning with Depthwise Separable Convolutions.” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017* 2017-Janua: 1800–1807.

Dai, W., Q. Yang, X. Gui-Rong, and Y. Yong. 2007. “Boosting for Transfer Learning

Wenyuan." *Proceedings of the 24th International Confer- ence on Machine Learning*: 193–200.

Dimililer, Kamil, Buse Ugur, and Yoney K. Ever. 2017. "Tumor Detection on Ct Lung Images Using Image." *The Online Journal of Science and Technology* 7(1): 2013–18.

"F127 70 : 2 @ 70 017 D 2Dg : // H / 2 # Ccc." 4: 5–9.

Gebremariam, Tewodros H. et al. 2021. "Clinical Characteristics and Molecular Profiles of Lung Cancer in Ethiopia." *JTO Clinical and Research Reports* 2(7): 100196. https://doi.org/10.1016/j.jtocrr.2021.100196.

Gomes, Rahul et al. 2022. "A Comprehensive Review of Machine Learning Used to Combat COVID-19." *Diagnostics* 12(8).

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. "Deep Residual Learning for Image Recognition." *Engineering Access* 7(1): 50–60.

Heuvelmans, Marjolein A. et al. 2021. "Lung Cancer Prediction by Deep Learning to Identify Benign Lung Nodules." *Lung Cancer* 154(November 2020): 1–4. https://doi.org/10.1016/j.lungcan.2021.01.027.

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. 2017. "Densely Connected Convolutional Networks." *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017* 2017-Janua: 2261–69.

Humayun, Mamoona, R. Sujatha, Saleh Naif Almuayqil, and N. Z. Jhanjhi. 2022. "A Transfer Learning Approach with a Convolutional Neural Network for the Classification of Lung Carcinoma." *Healthcare (Switzerland)* 10(6): 1–15.

Ji, Junyi. 2019. "Gradient-Based Interpretation on Convolutional Neural Network for Classification of Pathological Images." *Proceedings - 2019 International Conference on Information Technology and Computer Application, ITCA 2019*: 83–86.

Jin, Jill. 2021. "Screening for Lung Cancer." *JAMA - Journal of the American Medical Association* 325(10): 1016.

Kapileswar, N. et al. 2022. "Deep Learning Algorithm for Classification and Prediction of

COVID-19 Pneumonia Using CT Scan Images." *AIP Conference Proceedings* 2405.

Kareem, Hamdalla F. et al. 2021. "Evaluation of SVM Performance in the Detection of Lung Cancer in Marked CT Scan Dataset." *Indonesian Journal of Electrical Engineering and Computer Science* 21(3): 1731–38.

Kim, Hee E et al. 2022. "Transfer Learning for Medical Image Classification : A Literature Review." *BMC Medical Imaging*: 1–13. https://doi.org/10.1186/s12880-022-00793-7.

Kowsari, Kamran et al. "HMIC: Hierarchical Medical Image Classification, A Deep Learning Approach."

Krotkov, Eric, and John Blitch. 1999. "<Krotkov_1999_The Defense Advanced Research Projects Agency (DAR.Pdf>."

LaLonde, Rodney et al. 2021. "Capsules for Biomedical Image Segmentation." *Medical Image Analysis* 68: 101889. https://doi.org/10.1016/j.media.2020.101889.

Latif, Jahanzaib, Chuangbai Xiao, Azhar Imran, and Shanshan Tu. 2019. "Medical Imaging Using Machine Learning And." *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*: 1–5.

Lecun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. "Deep Learning." *Nature* 521(7553): 436–44.

Mali Patil et al. 2023. "Early Detection of Dementia Using Deep Learning and Image Processing." *International Journal of Engineering and Manufacturing* 13(1): 14–22.

Mehmood, Abid. 2021. "Efficient Anomaly Detection in Crowd Videos Using Pre-Trained 2D Convolutional Neural Networks." *IEEE Access* 9(October): 138283–95.

Minna, John D., Jack A. Roth, and Adi F. Gazdar. 2002. "Focus on Lung Cancer." *Cancer Cell* 1(1): 49–52.

Panwar, Harsh, P K Gupta, Mohammad Khubeb, and Ruben Morales-menendez. 2020. "Since January 2020 Elsevier Has Created a COVID-19 Resource Centre with Free Information in English and Mandarin on the Novel Coronavirus COVID- 19 . The COVID-19 Resource Centre Is Hosted on Elsevier Connect , the Company ' s Public News and

Information .” (January).

Rajasekar, Vani et al. 2023. “Lung Cancer Disease Prediction with CT Scan and Histopathological Images Feature Analysis Using Deep Learning Techniques.” *Results in Engineering* 18(August 2022): 101111. https://doi.org/10.1016/j.rineng.2023.101111.

Rajpurkar, Pranav et al. 2017. “CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning.” : 3–9. http://arxiv.org/abs/1711.05225.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. “‘Why Should i Trust You?’ Explaining the Predictions of Any Classifier.” *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 13-17-Augu: 1135–44.

Rong, Z. H.U., D. A.I. Lingyun, L. I.U. Jinxing, and G. U.O. Ying. 2021. “Diagnostic Classification of Lung Cancer Using Deep Transfer Learning Technology and Multi-Omics Data.” *Chinese Journal of Electronics* 30(5): 843–52.

Schabath, Matthew B., and Michele L. Cote. 2019. “Cancer Progress and Priorities: Lung Cancer.” *Cancer Epidemiology Biomarkers and Prevention* 28(10): 1563–79.

Selvaraju, Ramprasaath R. et al. 2017. “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization.” *Proceedings of the IEEE International Conference on Computer Vision* 2017-Octob: 618–26.

———. 2019. “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization.” *International Journal of Computer Vision* 128(2): 336–59.

Sfeir, Agnel and Symington, Lorraine S. 2017. “HHS Public Access.” *Physiology & behavior* 176(3): 139–48.

Shafi, Imran et al. 2022. “An Effective Method for Lung Cancer Diagnosis from CT Scan Using Deep Learning-Based Support Vector Network.” *Cancers* 14(21): 1–18.

Shaha, Manali. 2018. “Transfer Learning for Image Classification.” (Iceca): 656–60.

Shakeel, P. Mohamed, M. A. Burhanuddin, and Mohammad Ishak Desa. 2022. “Automatic Lung Cancer Detection from CT Image Using Improved Deep Neural Network and

Ensemble Classifier." *Neural Computing and Applications* 34(12): 9579–92. https://doi.org/10.1007/s00521-020-04842-6.

Shen, Wei et al. 2015. "Multi-Scale Convolutional Neural Networks for Lung Nodule Classification." *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9123: 588–99.

Shrestha, Ajay, and Ausif Mahmood. 2019. "Review of Deep Learning Algorithms and Architectures." *IEEE Access* 7: 53040–65.

Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman. 2014. "Deep inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps." *2nd International Conference on Learning Representations, ICLR 2014 - Workshop Track Proceedings*: 1–8.

Singh, Gur Amrit Pal, and P. K. Gupta. 2019. "Performance Analysis of Various Machine Learning-Based Approaches for Detection and Classification of Lung Cancer in Humans." *Neural Computing and Applications* 31(10): 6863–77. https://doi.org/10.1007/s00521-018-3518-x.

Van der Straeten, M., and J. Tasson. 1985. "Lung Cancer: Detection, Diagnosis and Staging." *Journal Belge de Radiologie* 68(3): 199–204.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. 2017. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning."

Thi, Phuong et al. 2024. "The Application of Deep Learning in Lung Cancerous Lesion Detection." : 1–21.

Travis, William D. 2011. "Pathology of Lung Cancer." *Clinics in Chest Medicine* 32(4): 669–92. http://dx.doi.org/10.1016/j.ccm.2011.08.005.

van der Velden, Bas H.M., Hugo J. Kuijf, Kenneth G.A. Gilhuijs, and Max A. Viergever. 2022. "Explainable Artificial Intelligence (XAI) in Deep Learning-Based Medical Image Analysis." *Medical Image Analysis* 79: 102470. https://doi.org/10.1016/j.media.2022.102470.

Venkatesh, C. et al. 2023. “A Hybrid Model for Lung Cancer Prediction Using Patch Processing and Deeplearning on CT Images.” *Multimedia Tools and Applications* (0123456789). https://doi.org/10.1007/s11042-023-17349-8.

Wang, Chi Chung et al. 2015. “HOXA5 Inhibits Metastasis via Regulating Cytoskeletal Remodelling and Associates with Prolonged Survival in Non-Small-Cell Lung Carcinoma.” *PLoS ONE* 10(4): 1–20.

Windisch, Paul et al. 2020. “Implementation of Model Explainability for a Basic Brain Tumor Detection Using Convolutional Neural Networks on MRI Slices.” *Neuroradiology* 62(11): 1515–18.

Xie, Saining, Ross Girshick, and Piotr Doll. 2017. “Aggregated Residual Transformations for Deep Neural Networks Http://Arxiv.Org/Abs/1611.05431v2.” *Conference on Computer Vision and Pattern Recognition*: 1492–1500.

Yan, Jingxin et al. 2023. “Efficacy and Complications of Transarterial Chemoembolization Alone or in Combination with Different Protocols for Hepatocellular Carcinoma: A Bayesian Network Meta-Analysis of Randomized Controlled Trials.” *iLIVER* 2(2): 130–41. https://doi.org/10.1016/j.iliver.2023.03.002.

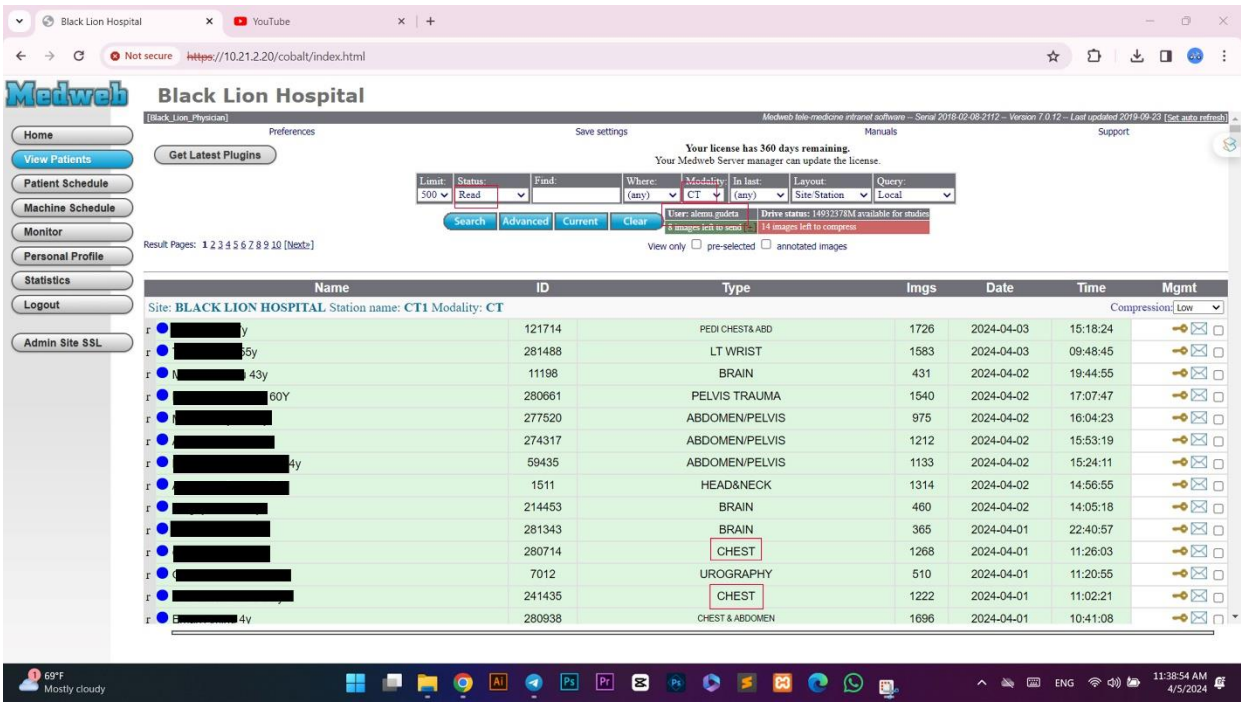
Yang, Xiangli et al. 2013. “A Survey on Deep Semi-Supervised Learning.” : 1–26.

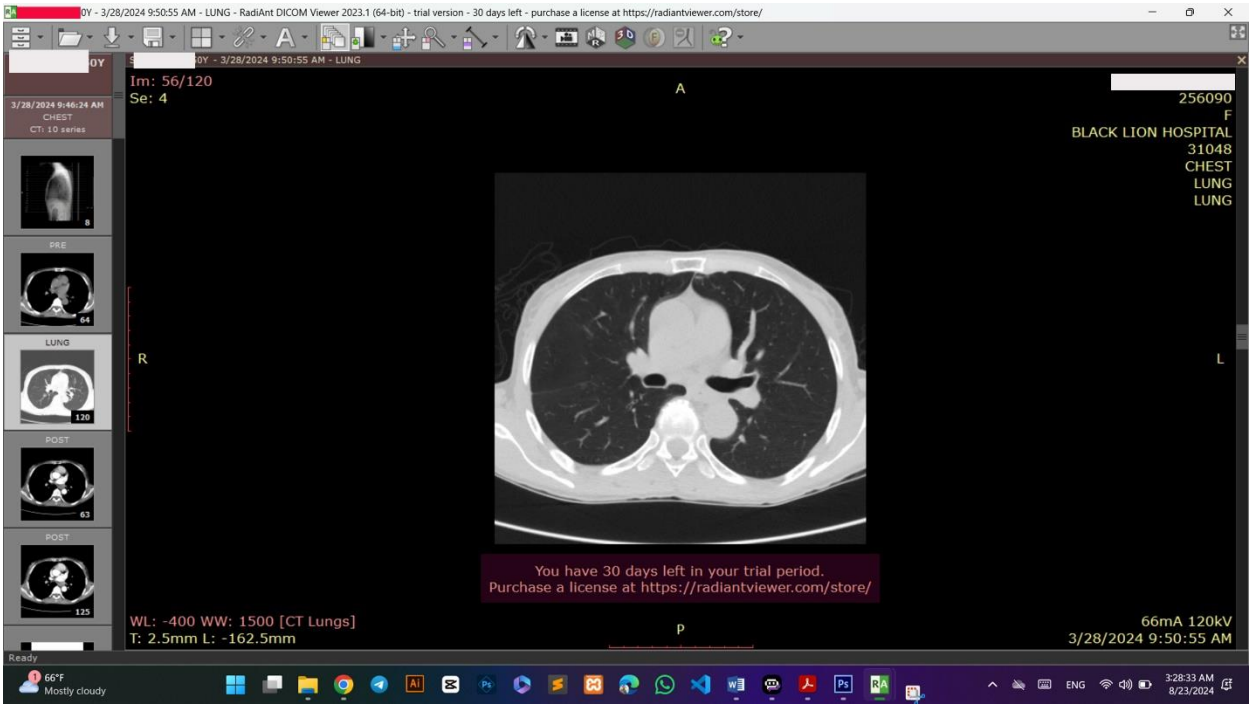
Zhou, Bolei et al. 2016. “Learning Deep Features for Discriminative Localization.” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2016-Decem: 2921–29.

Appendices

A. Data Collection Evidence

Disclaimer: Screenshot are manipulated to hide Privacy information, i.e. patient's name.





B. Full Grad-CAM implementation using python

```
import numpy as np
import matplotlib.pyplot as plt
import cv2
import tensorflow as tf
import keras
from keras import backend as K
from keras.models import load_model
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
tf.compat.v1.disable_eager_execution()
import argparse # Import the argparse module
```

```
import argparse
from keras.models import load_model
args_list = [
    '--base', 'dense',
    '--model_path', 'D:/GradCAM/code2stores/model/dense169.h5',
    '--input_size', '224',
    '--img', 'D:/GradCAM/mydata2/Testing/Cancer/Cancer (565).png',
    '--layer', 'conv5_block18_1_conv', # conv5_block32_2_conv
    '--out', '/content/drive/MyDrive/mydata/GradCamResult.png'
]
```

```
parser = argparse.ArgumentParser()
parser.add_argument('--base', type=str, help='Base model to use')
parser.add_argument('--model_path', type=str, help='Path to the saved model')
parser.add_argument('--input_size', type=int, help='Input size for the model')
parser.add_argument('--img', type=str, help='Path to the image to process')
parser.add_argument('--layer', type=str, help='Layer to visualize')
parser.add_argument('--out', type=str, help='Path to save the output image')

args = parser.parse_args(args_list)
print(args.base)
```

```
base = args.base
SIZE = args.input_size
path = args.img
out_dir = args.out
layer_name = args.layer
model = load_model(args.model_path)
```

```
def process(path, base, SIZE):
    x = keras.utils.load_img(path, target_size=(SIZE, SIZE))
    x = keras.utils.img_to_array(x)
    if base in ['vgg16', 'vgg19', 'incepv3', 'xcep']:
        x = x / 255.0
    elif base in ['res50', 'res101']:
        x -= np.mean(x, axis=(0, 1, 2))
        x = x[..., ::-1]
    elif base in ['dense', 'mob', 'incepres']:
        x = (x / 127.5) - 1.0
    else:
        raise ValueError('Invalid model specified')
    x = np.expand_dims(x, axis=0)

    return x
```

```
    fig, ax = plt.subplots(1, 2, figsize=(12, 6))
    ax[0].imshow(keras.utils.load_img(path), cmap='gray')
    ax[0].set_title('Original Image')
    ax[0].axis('off')
    ax[1].imshow(keras.utils.load_img(path), cmap='gray')
    ax[1].imshow(cam, cmap='jet', alpha=0.5)
    ax[1].set_title('GradCAM Output')
    ax[1].axis('off')
    plt.show()
```

```
def main(path, base, SIZE, model, layer_name):
    labels = {0: 'NonCancer', 1: 'Cancer'}
    truth = 1 # index of class 'Cancer'
    x_ = keras.utils.load_img(path)
    x_ = keras.utils.img_to_array(x_)
    x = process(path, base, SIZE)
    pred = model.predict(x)
    pred_index = np.argmax(pred)
    pred_label = labels[pred_index]
    pred_accuracy = pred[0, pred_index] * 100
    print(f'Ground truth: {truth} - {labels[truth]}')
    print(f'Predicted: {pred_index} - {pred_label} (accuracy: {pred_accuracy:.2f}%)')

    output_mod = model.output
    output_all_cats = output_mod[0]
    y_c = output_all_cats[pred_index]
    spatial_map_layer = model.get_layer(layer_name).output
```

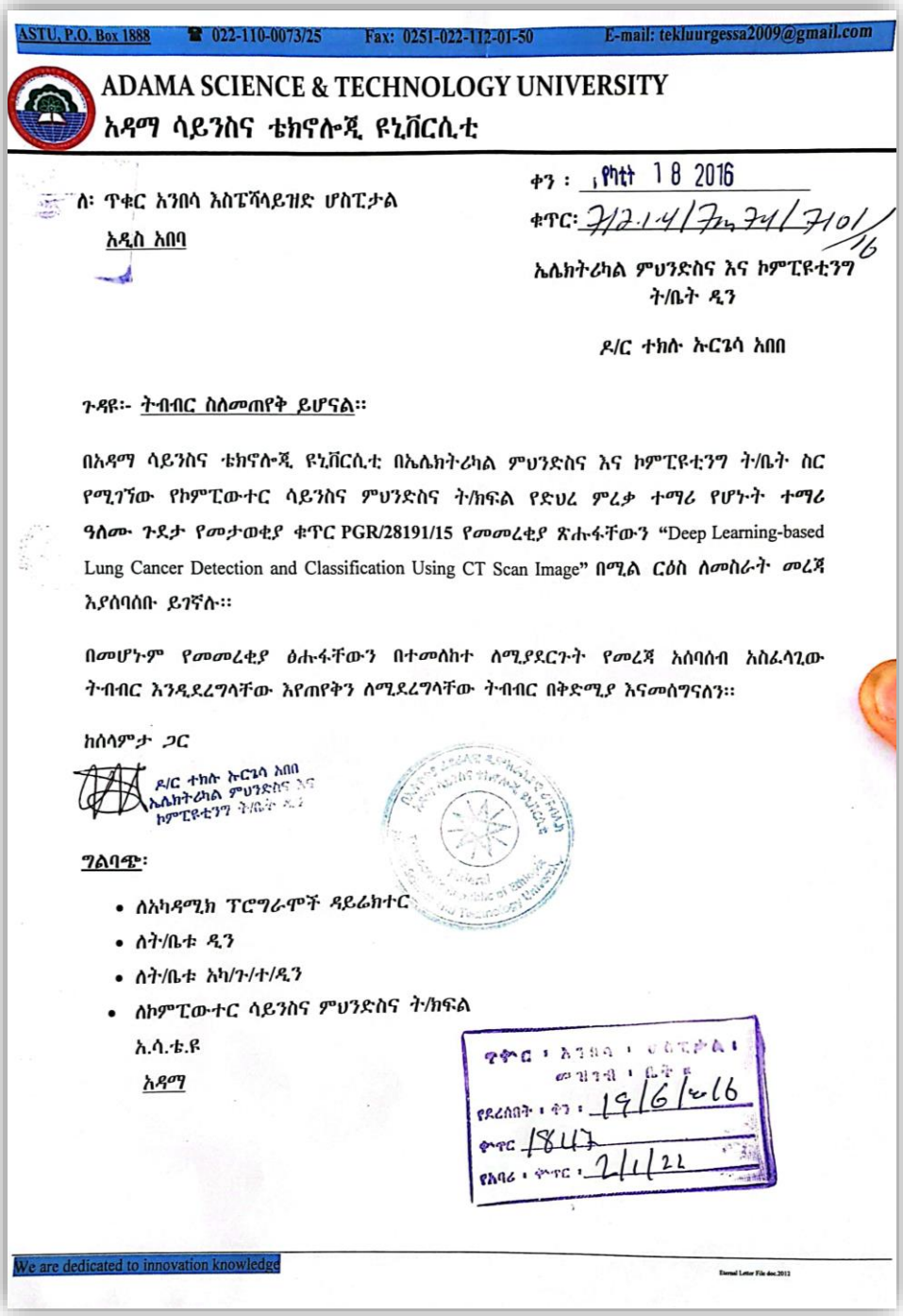
```
    grads_l = K.gradients(y_c, spatial_map_layer)
    grads = grads_l[0]
    spatial_map_and_gradient_function = K.function
                                        ([model.input], [spatial_map_layer, grads])
    spatial_map_all_dims, grads_val_all_dims = spatial_map_and_gradient_function([x])
    spatial_map_val = spatial_map_all_dims[0]
    grads_val = grads_val_all_dims[0]
    weights = np.mean(grads_val, axis=(0,1))

    cam = np.dot(spatial_map_val, weights)
    cam = np.maximum(cam, 0)
    cam = cv2.resize(cam, (x_.shape[1], x_.shape[0]), cv2.INTER_AREA)
    cam = cam / cam.max()

    fig, ax = plt.subplots(1, 2, figsize=(8, 5))
    ax[0].imshow(keras.utils.load_img(path), cmap='gray')
    ax[0].set_title('Original Image')
   ax[0].axis('off')
   ax[1].imshow(keras.utils.load_img(path), cmap='gray')
   ax[1].imshow(cam, cmap='jet', alpha=0.5)
   ax[1].set_title('GradCAM Output')
   ax[1].axis('off')
   plt.show()

if __name__ == '__main__':
    main(path, base, SIZE, model, layer_name)
```

C. Data Collection Request and Permission Letters



ASTU, P.O. Box 1888 ☎ 022-110-0073/25 Fax: 0251-022-112-01-50 E-mail: tekluurgessa2009@gmail.com

ADAMA SCIENCE & TECHNOLOGY UNIVERSITY
አዳማ ሳይንስና ቴክኖሎጂ ዩኒቨርሲቲ

ለ፡ ጥቁር አንበሳ እስፔሻላይዝድ ሆስፒታል
አዲስ አበባ

ቀን : ሚያዚያ 18 2016
ቁጥር፡ ኢ.ሳ.ቴ.ዩ/ኤሌ.ኮ/ት/ዲን/3101/16

ኤሌክትሪካል ምህንድስና እና ኮምፒዩቲንግ ት/ቤት ዲን

ዶ/ር ተክሉ ኡርጌሳ አበበ

ጉዳዩ፡- ትብብር ስለመጠየቅ ይሆናል።

በአዳማ ሳይንስና ቴክኖሎጂ ዩኒቨርሲቲ በኤሌክትሪካል ምህንድስና እና ኮምፒዩቲንግ ት/ቤት ስር የሚገኘው የኮምፒውተር ሳይንስና ምህንድስና ት/ክፍል የድህረ ምረቃ ተማሪ የሆኑት ተማሪ ዓለሙ ጉደታ የመታወቂያ ቁጥር PGR/28191/15 የመመረቂያ ጽሑፋቸውን "Deep Learning-based Lung Cancer Detection and Classification Using CT Scan Image" በሚል ርዕስ ለመስራት መረጃ እያሰባሰቡ ይገኛሉ።

በመሆኑም የመመረቂያ ፅሑፋቸውን በተመለከተ ለሚያደርጉት የመረጃ አሰባሰብ አስፈላጊው ትብብር እንዲደረግላቸው እየጠየቅን ለሚደረግላቸው ትብብር በቅድሚያ እናመሰግናለን።

ከሰላምታ ጋር

ዶ/ር ተክሉ ኡርጌሳ አበበ
ኤሌክትሪካል ምህንድስና እና ኮምፒዩቲንግ ት/ቤት ዲን

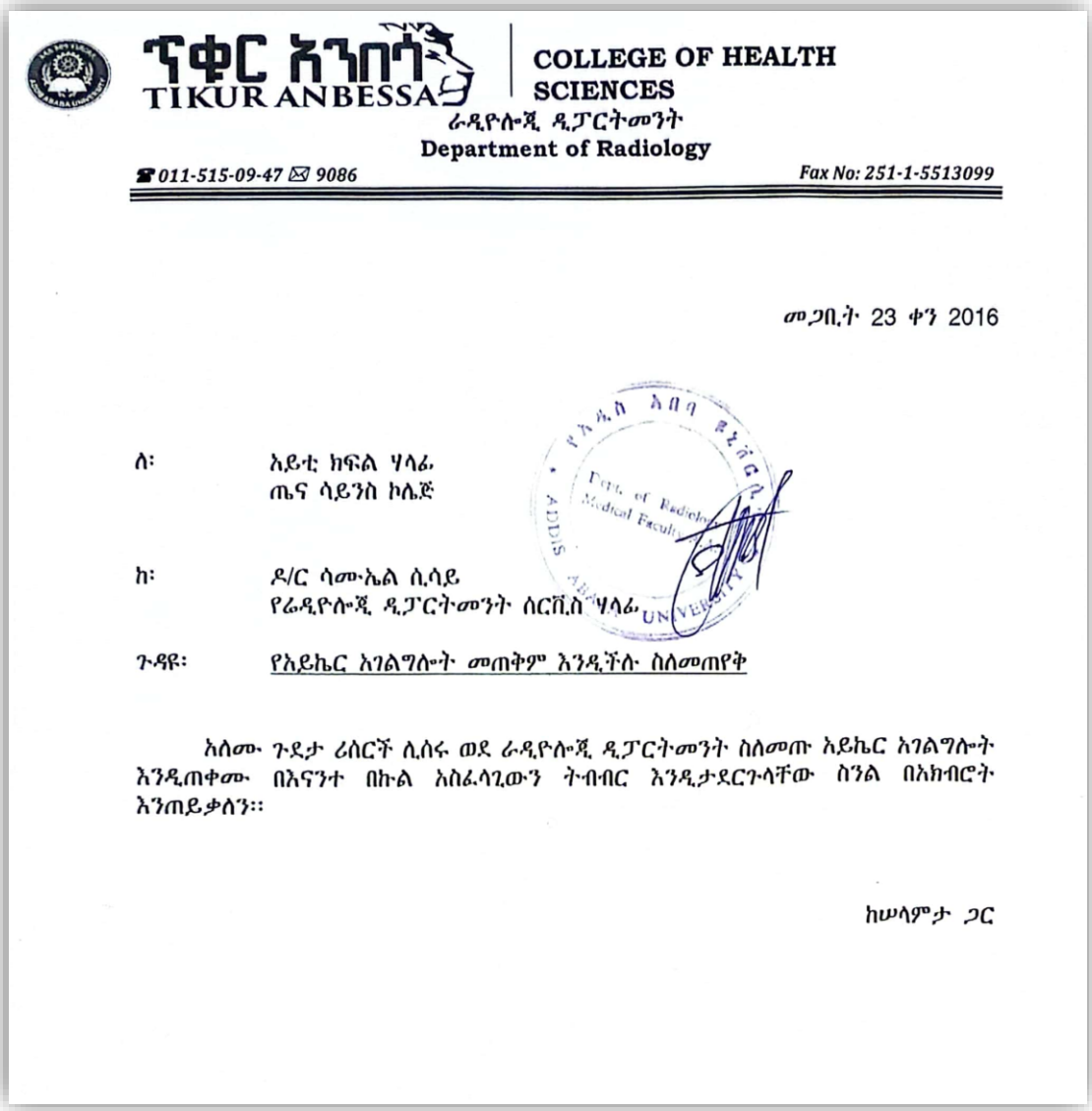
ግልባጭ፡
- ለአካዳሚክ ፕሮግራሞች ዳይሬክተር
- ለት/ቤቱ ዲን
- ለት/ቤቱ አካ/ን/ተ/ዲን
- ለኮምፒውተር ሳይንስና ምህንድስና ት/ክፍል

አ.ሳ.ቴ.ዩ
አዳማ

ጥቁር አንበሳ ... ሆስፒታል ... ቤተ ... የደረሰበት ቀን : 19/6/16 ቁጥር 18447 የአባሪ ቁጥር : 2/1/22

We are dedicated to innovation knowledge

Description: The School of Electrical Engineering and Computing provided us with a letter to Tikur Anbessa Hospital for the purpose of gathering data at our request.



ጥቁር አንበሳ
TIKUR ANBESSA
COLLEGE OF HEALTH SCIENCES
ራዲዮሎጂ ዲፓርትመንት
Department of Radiology
☎ 011-515-09-47 ✉ 9086
Fax No: 251-1-5513099

መጋቢት 23 ቀን 2016

ለ፡ አይቲ ክፍል ሃላፊ
ጤና ሳይንስ ኮሌጅ

ከ፡ ዶ/ር ሳሙኤል ሲሳይ
የራዲዮሎጂ ዲፓርትመንት ሰርቪስ ሃላፊ

ጉዳዩ፡ የአይኬር አገልግሎት መጠቅም እንዲችሉ ስለመጠየቅ

ለመሙ ጉደታ ሪሰርች ሊሰሩ ወደ ራዲዮሎጂ ዲፓርትመንት ስለመጡ አይኬር አገልግሎት እንዲጠቀሙ በእናንተ በኩል አስፈላጊውን ትብብር እንዲታደርጉላቸው ስንል በአክብሮት እንጠይቃለን፡፡

ከሠላምታ ጋር

Description: Our request was granted by the Radiology department, and our process of obtaining MedWeb credentials to access data was then passed to the IT department. The official server and viewer for patient records at Tikur Anbessa Hospital is called MedWeb.

D. Training Plots with Full Layers of Pretrained models.

Note: When we disable early stopping and use all layers of the pretrained model, as shown in these plots and graphs, the model experiences overfitting and underfitting, which causes both training and validation accuracy to be 100 at epoch 50, 50, and even 10, or validation accuracy shows inconsistent and worse results. This is the rationale behind the significance of freezing a few model layers.